# Netra™ CT 900 Server Systemadministrationshandbuch

Sun Microsystems, Inc.
www.sun.com

Teilenummer: 820-0570-10
Januar 2007, Version A

Bitte senden Sie Ihre Anmerkungen zu diesem Dokument an: `http://www.sun.com/hwdocs/feedback`

Adobe PostScript

# Inhalt

# Abbildungen

# Tabellen

# Vorwort

Das *Netra CT 900 Server Systemadministrationshandbuch* enthält Konfigurations- und Administrationsinformationen für Systemadministratoren des Netra™ CT 900 Servers. Außerdem enthält das Handbuch Informationen zum Shelf Manager und eine IPMI-Befehlsreferenz.

In diesem Handbuch wird vorausgesetzt, dass Sie mit UNIX®-Befehlen und -Netzwerken, den Grundlagen der PICMG® 3.x AdvancedTCA®-Spezifikation und dem Intelligent Platform Management Interface (IPMI) vertraut sind.

## Aufbau dieses Handbuchs

Kapitel 1 enthält eine Einführung in die Netra CT 900 Server-Software.

Kapitel 2 enthält Informationen zur Konfiguration des Systems.

Kapitel 3 enthält eine Beschreibung der Systemadministration.

Anhang A enthält die Syntax und Anwendungsmöglichkeiten der Shelf Manager-CLI-Befehle (Command-Line Interface = Befehlszeilenschnittstelle).

Anhang B enthält eine Beschreibung der Sun-spezifischen, OEM-definierten IPMI-Befehle (Intelligent Platform Management Interface).

# Verwenden von UNIX-Befehlen

Dieses Dokument enthält unter Umständen keine Informationen zu grundlegenden UNIX-Befehlen und -Verfahren (z. B. das Herunterfahren oder Starten des Systems und das Konfigurieren von Geräten). Weitere Informationen hierzu finden Sie unter:

- Software-Dokumentation, die Sie mit Ihrem System erhalten haben
- Dokumentation zum Solaris™-Betriebssystem (Solaris OS) unter der folgenden Adresse:

  `http://docs.sun.com`

# Shell-Eingabeaufforderungen

| Shell | Eingabeaufforderung |
| --- | --- |
| C-Shell | *computername*% |
| C-Shell-Superuser | *computername*# |
| Bourne-Shell und Korn-Shell | $ |
| Bourne-Shell- und Korn-Shell-Superuser | # |

# Typografische Konventionen

| Schriftart* | Bedeutung | Beispiele |
|---|---|---|
| `AaBbCc123` | Namen von Befehlen, Dateien und Verzeichnissen; Bildschirmausgaben | Bearbeiten Sie die `.login`-Datei.<br>Mithilfe von `ls -a` können Sie alle Dateien auflisten.<br>`% Sie haben Post.` |
| **`AaBbCc123`** | Tastatureingaben im Gegensatz zu Bildschirmausgaben des Computers | `%` **`su`**<br>`Password:` |
| *AaBbCc123* | Buchtitel, neu eingeführte Begriffe oder Betonungen. Ersetzen Sie die Befehlszeilenvariablen durch echte Namen oder Werte. | Siehe Kapitel 6 im *Benutzerhandbuch*.<br>Diese Optionen werden als *Klassenoptionen* bezeichnet.<br>Dazu *müssen* Sie als Superuser angemeldet sein.<br>Geben Sie zum Löschen einer Datei `rm` *Dateiname* ein. |

* Ihr Browser verwendet möglicherweise andere Einstellungen.

# Zugehörige Dokumentation

Die gesamte Netra CT 900 Server-Dokumentation ist in der folgenden Tabelle aufgeführt. Mit Ausnahme der Dokumentation *Important Safety Information for Sun Hardware Systems* sind alle aufgeführten Dokumente unter der folgenden Adresse online verfügbar:

`http://www.sun.com/documentation`

| Titel | Teilenummer |
|---|---|
| *Netra CT 900 Server Product Notes* | 819-1180 |
| *Netra CT 900 Server – Überblick* | 820-0554 |
| *Netra CT 900 Server Installationshandbuch* | 820-0562 |
| *Netra CT 900 Server Service Manual* | 819-1176 |

| Titel | Teilenummer |
|---|---|
| *Netra CT 900 Server Switch Software Reference Manual* | 819-3774 |
| *Netra CT 900 Server Safety and Compliance Manual* | 819-1179 |
| *Important Safety Information for Sun Hardware Systems (nur als Druckversion verfügbar)* | 816-7190-10 |

Weitere Informationen finden Sie bei Bedarf in den Dokumentationen zu den folgenden Produkten: Solaris OS, OpenBoot™ PROM Firmware, Netra CP3010 Board, Netra CP3020 Board und Netra CP3060 Board.

# Dokumentation, Support und Schulung

| Sun-Funktion | URL |
|---|---|
| Dokumentation | `http://www.sun.com/documentation/` |
| Support | `http://www.sun.com/support/` |
| Schulung | `http://www.sun.com/training/` |

# Kontaktieren der technischen Unterstützung von Sun

Bei technischen Fragen zu diesem Produkt, die in diesem Dokument nicht beantwortet werden, finden Sie weitere Informationen unter:

`http://www.sun.com/service/contacting`

# Kommentare und Anregungen

Wir bemühen uns um eine stetige Verbesserung unserer Dokumentation und freuen uns über Ihre Kommentare und Anregungen. Senden Sie uns Ihre Kommentare unter:

`http://www.sun.com/hwdocs/feedback`

Bitte geben Sie dabei den Titel und die Teilenummer Ihres Dokuments an:

*Netra CT 900 Server Systemadministrationshandbuch*, Teilenummer 820-0570-10.

KAPITEL 1

# Einführung

Dieses Kapitel enthält folgende Abschnitte:



## Netra CT 900 Server-Software

Die Netra CT 900 Server-Software umfasst Folgendes:

- Shelf Manager
- Betriebssysteme und Anwendungen
- Firmware

---

**Hinweis –** In der Advanced Telecom Computing Architecture® (ATCA) wurde der Ausdruck *Shelf* aus der Telekommunikationsterminologie übernommen. Früher wurde stattdessen der Ausdruck *Chassis* verwendet, der praktisch dieselbe Bedeutung hat.

---

Die Software ist in TABELLE 1-1 beschrieben und in ABBILDUNG 1-1 im logischen Zusammenhang mit der Hardware dargestellt.

**TABELLE 1-1** Netra CT Server-Software für Systemadministratoren

| Kategorie | Name | Beschreibung |
|---|---|---|
| *Shelf-Verwaltung* | IPM Sentry-Shelf Manager | Die Shelf Manager-Software wird über die Shelf-Verwaltungskarte (ShMM) ausgeführt und ist werkseitig vorinstalliert. Sie stellt das Remote Management Control Protocol (RMCP) und den CLI-Zugriff auf das IPMI zum Verwalten des Servers bereit. |
| | Command-Line Interface (CLI) | Das CLI ist eine integrierte Benutzeroberfläche für den Shelf Manager. |
| *Betriebssysteme und Anwendungen* | Solaris Operating System (Solaris OS) | Das Solaris OS wird auf ATCA-kompatiblen Node Boards mit Sun-Unterstützung ausgeführt, z. B. auf den Node Boards Netra CP3010, Netra CP3020 und CP3060. Solaris 10 ist auf Wunsch auf den Netra Node Boards vorinstalliert. Benutzer können Solaris 10 und andere Solaris OS-Versionen auch herunterladen und installieren. |
| | Monta Vista Carrier Grade Linux OS | Das Netra CP3020 kann auch das Monta Vista Carrier Grade Linux OS ausführen. |
| *Firmware* | OpenBoot PROM-Firmware | Firmware auf Node Boards mit Sun-Unterstützung, die das Starten steuert, z. B. das Netra CP3010 Board. Die Firmware enthält auch Diagnosefunktionen. |
| | U-Boot | Firmware auf den Shelf-Verwaltungskarten, die einen Selbsttest beim Einschalten (Power-On Self-Test, POST) durchführt und das Starten der Software für die Shelf-Verwaltungskarte steuert. |
| | Intelligent Platform Management Controller (IPMC) | Systemverwaltungscontroller-Firmware, die auf einem Node Board mit Sun-Unterstützung die Kommunikation über den IPMI-Controller ermöglicht. |

Der Netra CT 900 Server verfügt über zwei Shelf-Verwaltungskarten (ShMMs) und bietet für bestimmte Hardware- und Software-Ereignisse ein Shelf-Verwaltungskarten-Failover von der aktiven Shelf-Verwaltungskarte zur Standby-Shelf-Verwaltungskarte. Die *aktive* Shelf-Verwaltungskarte wird auf Systemebene für die Konfiguration, Administration und Verwaltung der meisten Komponenten verwendet, die an die Mittelplatine angeschlossen sind. Die *Standby*-Shelf-Verwaltungskarte stellt Redundanz- und Failoverfunktionen für die aktive Shelf-Verwaltungskarte bereit.

Die Switching Fabric Boards sorgen für die interne Verbindung der Shelf-Verwaltungskarte und der Node Boards und verfügen an der Rückseite über Ethernet-Ports für externe Verbindungen.

Die Netra CP3010 Node Boards können in Verbindung mit Peripheriegeräten betrieben werden, z. B. Festplatten. Außerdem können auf Node Boards Benutzeranwendungen ausgeführt werden. Bei einem Netra CT 900 Server führt jedes Node Board seine eigene Betriebssysteminstanz aus und kann daher jeweils als Server bezeichnet werden. Die Shelf-Verwaltungskarten, Node Boards, Switching Fabric Boards und die anderen Field Replaceable Units (FRUs = austauschbare Teile bzw. Module) bilden ein System.

---

**Hinweis –** Wenn keine anderslautenden Angaben gemacht werden, wird der Ausdruck *Node Board* in diesem Handbuch für eine ATCA-CPU-Karte mit Sun-Unterstützung verwendet, z. B. das Netra CP3010 Board.

ATCA Node Boards von anderen Herstellern, die PICMG® 3.x-kompatibel sind, können im Netra CT 900 Server verwendet werden. Auf diesen Boards wird in der Regel kein Solaris OS ausgeführt, und die Systemverwaltungssoftware für den Netra CT 900 Server wird ebenfalls nicht ausgeführt. Aus diesem Grund ist im Vergleich zu Netra Node Boards nur eine eingeschränkte Verwaltung möglich.

---

TABELLE 1-2 enthält eine Übersicht über den Zugang zu den verschiedenen Boards. Die Shelf-Verwaltungskarte unterstützt 22 gleichzeitige Sitzungen (1 TIP- und 21 Telnet-Verbindungen).

**TABELLE 1-2** Zugangsverfahren für das Netra CT 900 System Board

| Board | Zugangsverfahren |
|---|---|
| Shelf-Alarmbedienfeld (SAP) | An der Frontblende befinden sich folgende Ports:<br>• Zwei serielle Ports (RS-232) mit RJ-45 DTE-Anschlüssen. Der serielle Port 1 ist die Konsolenverbindung für die obere Shelf-Verwaltungskarte (ShMM1). Dies ist standardmäßig die *aktive* Karte. Der serielle Port 2 ist die Konsolenverbindung für die untere Shelf-Verwaltungskarte (ShMM2). Dies ist standardmäßig die *Sicherungs*-Karte.<br>• Telco Alarm-Anschluss (DB-15) |
| Switchboard (Steckplätze 7 und 8) | Mehrere Ethernet-Ports für die Telnet-Verbindung an der Frontblende.<br>Hinweis – Für den Zugang zu diesen Ports von der Rückseite aus ist eine Rear Transition Card (RTC) erforderlich. Sie können entweder den Portanschluss an der Vorderseite oder an der Rückseite verwenden, jedoch nicht beide gleichzeitig. Wenn Sie an beide Ports ein Kabel anschließen, ist nur der Port an der Vorderseite aktiv. |
| Node Board (CPU-Boards mit Sun-Unterstützung) (Steckplätze 1 bis 6 und 9 bis 14) | Das Netra CP3010 Board verfügt an der Frontblende über die folgenden Ports:<br>• Zwei serielle Ports (Konsole) für TIP- oder ASCII-Terminal-Verbindungen<br>• Zwei Ethernet-Ports für Telnet-Verbindungen<br>• Einen 4X Serial Attached SCSI-Portanschluss (SAS)<br>Hinweis – Für den Zugang zu diesen Ports von der Rückseite aus ist eine RTC erforderlich. Sie können entweder den Portanschluss an der Vorderseite oder an der Rückseite verwenden, jedoch nicht beide gleichzeitig. Wenn Sie an beide Ports ein Kabel anschließen, ist nur der Port an der Vorderseite aktiv.<br>Das Netra CP3020 Board und das Netra CP3060 Board verfügen an der Vorderseite über die folgenden Ports:<br>• Einen seriellen Port (Konsole) für TIP- oder ASCII-Terminal-Verbindungen<br>• Zwei Ethernet-Ports für Telnet-Verbindungen<br>Weitere Informationen finden Sie in der Dokumentation zum Netra ATCA CPU Board des jeweiligen Node Boards. |
| Node Boards von anderen Herstellern (Steckplätze 1 bis 6 und 9 bis 14) | Je nach Board des jeweiligen Herstellers verschieden. |

Die Hardwareschnittstellen umfassen das Intelligent Platform Management Interface (IPMI), die Basisschnittstelle und die erweiterte Schnittstelle sowie die Netzwerkschnittstelle auf den Shelf-Verwaltungskarten, den Node Boards und den Switching Fabric Boards.

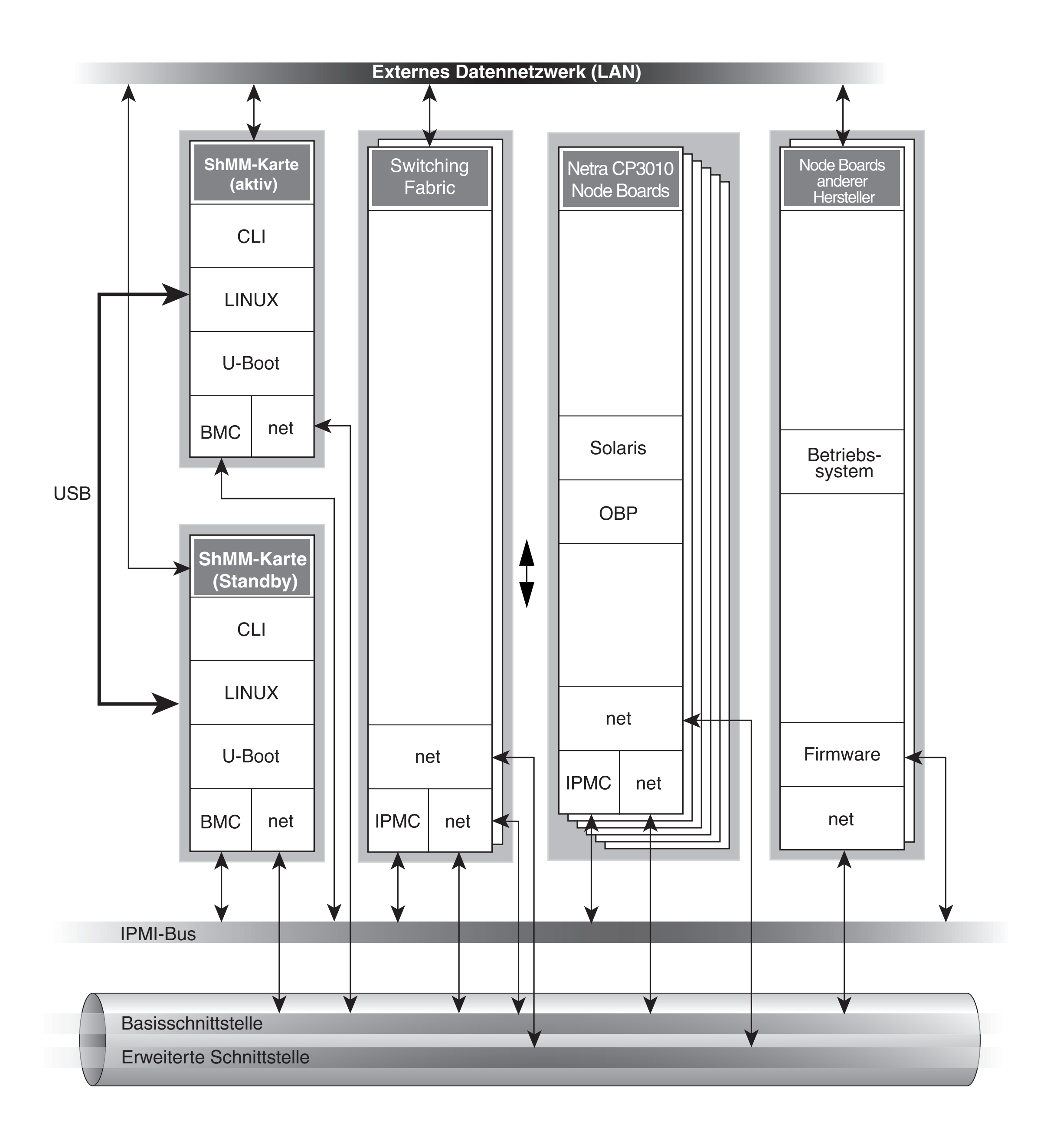



**ABBILDUNG 1-1** Logische Darstellung der Software- und Hardwareschnittstellen in einem Netra CT Server

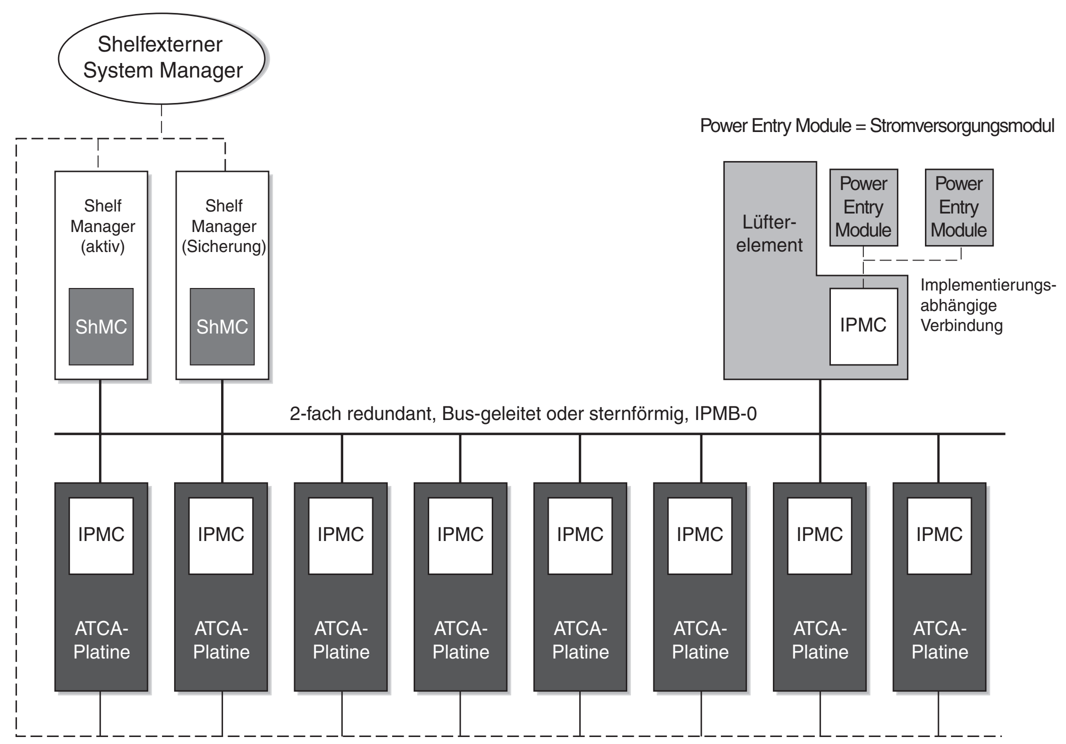
# Einführung in den Shelf Manager

Der Shelf Manager ist eine Verwaltungslösung für ATCA-Produkte auf Shelf-Ebene. Die Shelf-Verwaltungskarte stellt die Hardware bereit, die zum Ausführen des Shelf Managers in einem ATCA-Shelf erforderlich ist. In dieser Übersicht wird näher auf die Aspekte des Shelf Managers und der Shelf-Verwaltungskarte eingegangen, die in Verbindung mit ATCA für alle Carrier-Einheiten der Shelf-Verwaltung gelten.

## Überblick über Intelligent Platform Management in der ATCA

Beim Shelf Manager und der Shelf-Verwaltungskarte handelt es sich um Intelligent Platform Management-Komponenten (IPM), die für modulare Plattformen wie ATCA entwickelt wurden. Das Hauptaugenmerk liegt dabei auf dynamisch angeordneten FRUs und größtmöglicher Wartungsfreundlichkeit. Die IPMI-Spezifikation stellt ein solides Fundament für die Verwaltung von Plattformen dieser Art dar, aber für eine gute Unterstützung sind umfangreiche Erweiterungen erforderlich. Die erforderlichen Erweiterungen für das IPMI sind in PICMG 3.0, der ATCA-Spezifikation, definiert.

In ABBILDUNG 1-1 sind die logischen Elemente eines Beispiels für ein ATCA-Shelf anhand der ATCA-Spezifikation dargestellt.



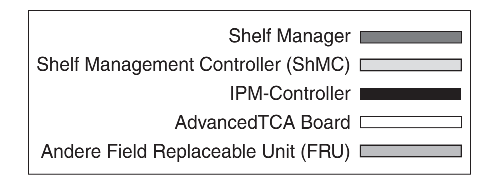




**ABBILDUNG 1-2** Beispiel für ein ATCA-Shelf

Ein übergeordneter System Manager (in der Regel shelfextern) kann die Aktivitäten mehrerer Shelves koordinieren. Eine System Manager-Einheit kommuniziert mit den einzelnen Shelf Manager-Instanzen meist über eine Ethernet-Schnittstelle oder serielle Schnittstelle.

In ABBILDUNG 1-2 sind die drei Verwaltungsebenen dargestellt: Board, Shelf und System. Im nächsten Abschnitt geht es um die Shelf Manager-Software und die Shelf-Verwaltungskarte, die verwendet werden, um einen ATCA-kompatiblen Shelf Manager und so genannten Shelf Management Controller (ShMC) zu implementieren.

## Shelf Manager und Shelf-Verwaltungskarte

Der Shelf Manager verfügt (in Übereinstimmung mit den ATCA-Anforderungen an Shelf Manager) über zwei Hauptzuständigkeiten:

- Verwalten und Überwachen der FRU-Elemente und allgemeinen Infrastruktur eines Shelfs, insbesondere die Einheiten für Stromversorgung, Lüftung und Verbindung sowie ihre Auslastung. Auf Shelf-Ebene erfolgt die Verwaltung und Überwachung vorrangig mithilfe von Interaktionen zwischen Shelf Manager und den IPM-Controllern über Intelligent Platform Management-Bus 0 (IPMB-0).
- Aktivieren der allgemeinen System Manager-Einheit, um die Verwaltung und Überwachung über die System Manager-Schnittstelle durchzuführen, die normalerweise per Ethernet implementiert wird.

Ein Großteil der Shelf Manager-Software beschäftigt sich mit Routineaufgaben wie dem Hoch- oder Herunterfahren eines Shelfs und dem Verwalten des Ein- und Ausbaus von FRUs. Dazu gehört auch die Zuweisung von Ressourcen für Stromversorgung und Verbindungen. Außerdem kann der Shelf Manager direkte Schritte einleiten, wenn auf Shelf-Ebene Ausnahmefehler auftreten. Der Shelf Manager kann bei Ausnahmebedingungen aufgrund der Temperatur z. B. die Lüftereinstellung erhöhen oder, falls diese Maßnahme nicht ausreicht, auch FRUs herunterfahren, um die Wärmebelastung im Shelf zu reduzieren.

## Shelf Manager-Leistungsmerkmale

Die Shelf Manager-Leistungsmerkmale umfassen Folgendes:

- Wird auf der Shelf-Verwaltungskarte ausgeführt (einem kompakten SO-DIMM-Formfaktormodul), die auf einer geeigneten Trägerkarte für das Shelf installiert ist.
- Erfüllt die ATCA-Spezifikation.
- Überwacht Aktivitäten auf Shelf-Ebene über den ATCA-spezifischen Intelligent Platform Management-Bus (IPMB) mit Zweifachredundanz.
- Akzeptiert und protokolliert Ereignisse, die von einer beliebigen intelligenten FRU auf Shelf-Ebene bereitgestellt werden (Ausnahmebedingungen der Temperatur, Spannung usw.); stellt Warnmeldungen außerhalb des Shelfs basierend auf konfigurierbaren Ereignisfiltern für die IPMI-Plattform bereit.
- Unterstützt den Austausch von Field Replaceable Units (FRUs) während des Betriebs unter Beibehaltung aller Verwaltungsfunktionen.
- Verfügt über eine Schnittstelle zu standardmäßigen Telco Alarm-Infrastrukturen, indem per Shelf-Verwaltungskarte implementierte Trockenkontaktrelais verwendet werden.
- Unterstützt redundante Shelf Manager-Instanzen, um eine hohe Verfügbarkeit zu erzielen.

- Umfasst einen Watchdog-Timer, der die Shelf-Verwaltungskarte zurücksetzt, wenn die regelmäßige Abtastung unterbrochen wird. Bei diesen Zurücksetzungen wird automatisch ein Switchover (Umschaltung) zur Sicherungs-Shelf-Verwaltungskarte ausgelöst, wenn dies entsprechend konfiguriert ist.
- Enthält eine akkugepufferte Echtzeituhr für Zeitstempelereignisse.
- Implementiert eine Vielzahl von externen Schnittstellen, die über Ethernet zugänglich sind (einschließlich RMCP, erforderlich für ATCA, und CLI).

## Shelf Manager-Switchover

Sie können den Shelf Manager mit aktiven und Sicherungsinstanzen konfigurieren, um die Verfügbarkeit zu erhöhen. In ABBILDUNG 1-3 ist dargestellt, wie beide Instanzen für den System Manager zugänglich sind, wobei nur mit der jeweils aktiven Instanz interagiert wird. Außerdem kommuniziert auch jeweils nur die aktive Instanz über den IPMB-0 mit den IPM-Controllereinheiten auf Shelf-Ebene. Die beiden Instanzen kommunizieren per TCP/IP miteinander, und die aktive Instanz stellt der Sicherungsinstanz inkrementelle Statusaktualisierungen bereit. Auf diese Weise kann die Sicherungsinstanz schnell die aktive Rolle übernehmen.

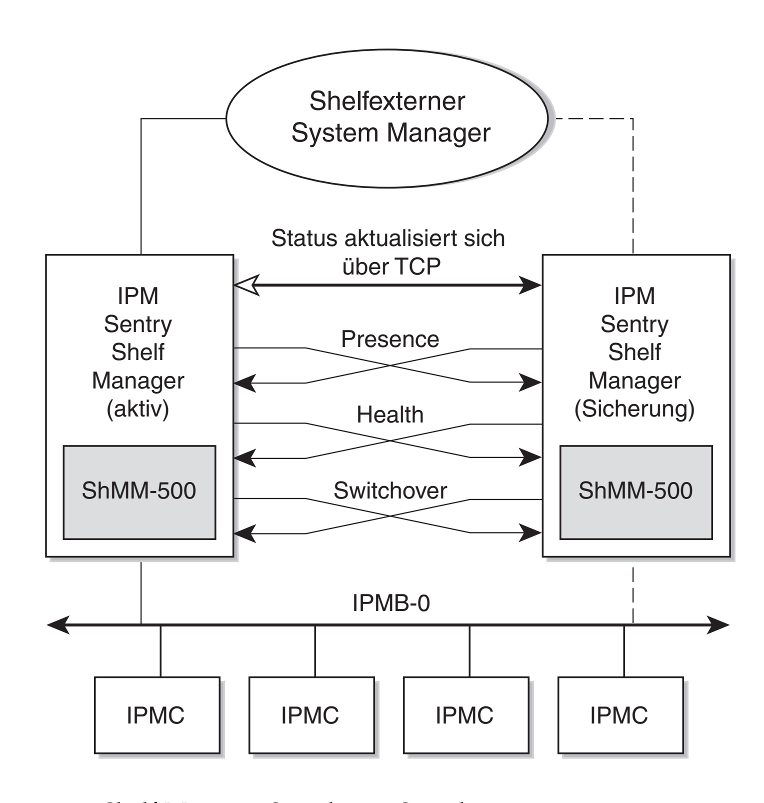



**ABBILDUNG 1-3** Shelf Manager-Switchover-Signale

TABELLE 1-3 enthält die Signale und deren Beschreibungen.

**TABELLE 1-3** Hardwaresignale und -schnittstellen mit Switchover-Unterstützung

| Hardware | Beschreibung |
|---|---|
| USB-Schnittstelle | Die Hauptschnittstelle zwischen den Shelf-Verwaltungskarten; wird verwendet, um sog. „Heartbeats“ und Informationen zur Statussynchronisierung zu senden. Beide Shelf-Verwaltungskarten müssen dieselbe FRU anzeigen, z. B. ein bestimmtes Lüfterelement oder ein Node Board in einem bestimmten Steckplatz, die denselben Status aufweisen, z. B. „Hochgefahren“. |
| #SWITCHOVER | Die Sicherungsinstanz kann bei Bedarf einen Switchover (Umschaltung) erzwingen. |
| #PRSNT | Dieses Signal zeigt das Vorhandensein einer Shelf-Verwaltungskarte an. |
| #HEALTHY | Dieses Signal zeigt die allgemeine Fehlerfreiheit der Shelf-Verwaltungskarte sowohl für die Hardware als auch für die Software an. |

## Switchover-Details

Der aktive Shelf Manager legt das ShMC-Gerät (Adresse 20h) auf dem IPMB offen, verwaltet den IPMB und die IPM-Controller und interagiert mit dem System Manager per RMCP und andere externe Schnittstellen. Er verfügt über eine offene TCP-Verbindung zum Sicherungs-Shelf Manager. Alle Änderungen am Status der verwalteten Objekte werden an den Sicherungs-Shelf Manager gemeldet.

Der Sicherungs-Shelf Manager legt den ShMC nicht auf dem IPMB offen, führt keine aktive Verwaltung des IPMB und der IPM-Controller durch und interagiert nicht über externe Schnittstellen mit dem System Manager (mit einer Ausnahme, die unten beschrieben ist). Stattdessen wird der Status der verwalteten Objekte im eigenen Speicher gepflegt (flüchtig und nicht-flüchtig) und auf Anweisung des aktiven Shelf Managers jeweils aktualisiert.

Der Sicherungs-Shelf Manager kann durch einen Switchover (Umschaltung) zur aktiven Einheit werden. Es sind zwei Arten von Switchovers (Umschaltungen) definiert:

- Kooperativer Switchover: Der aktive und der Sicherungs-Shelf Manager handeln die Übergabe von Zuständigkeiten vom aktiven Shelf Manager zum Sicherungs-Shelf Manager aus. Dieser Modus wird vom CLI-Befehl `switchover` unterstützt, der auf dem aktiven oder dem Sicherungs-Shelf Manager ausgeführt werden kann.
- Erzwungener Switchover: Der Sicherungs-Shelf Manager bestimmt, dass der aktive Shelf Manager nicht mehr aktiv ist bzw. fehlerfrei arbeitet und erzwingt die Übernahme der Zuständigkeiten vom aktiven Shelf Manager.

Der Sicherungs-Shelf Manager erkennt die Deaktivierung des aktiven Shelf Managers, wenn das Low-Level-Signal **Remote Healthy** oder **Remote Presence** inaktiv ist. Das Signal **Remote Presence** überwacht das Vorhandensein der gleichgeordneten Shelf Manager-Instanz. Wenn dieses Signal nicht mehr aktiv ist, bedeutet dies, dass das als Host für den gleichgeordneten Shelf Manager fungierende Board aus dem Shelf entfernt wurde. Das Signal **Remote Healthy** wird während der Initialisierung vom gleichgeordneten Shelf Manager eingerichtet. Wenn dieses Signal nicht mehr aktiv ist, bedeutet dies, dass der Remote-Shelf Manager nicht mehr fehlerfrei ausgeführt wird (in der Regel also heruntergefahren oder zurückgesetzt wurde).

Eine andere Situation, in der Aktionen des Sicherungs-Shelf Managers erforderlich sind, besteht, wenn die TCP-Verbindung zwischen den beiden Shelf Manager-Instanzen unterbrochen wird. Dies ist der Fall, wenn der Kommunikationslink zwischen den beiden Shelf Manager-Einheiten unterbrochen wird, wenn der Shelf Management-Prozess auf dem aktiven Shelf Manager beendet wird (entweder absichtlich oder unabsichtlich) oder wenn eine Softwareausnahme eintritt. Da die TCP-Option `keepalive` für die Verbindung aktiviert ist, wird sie kurz nach dem Ausschalten oder Zurücksetzen der aktiven Shelf-Verwaltungskarte geschlossen.

Bei einer Beendigung der Shelf Manager-Einheit ist es möglich, dass die TCP-Verbindung geschlossen wird, *bevor* das Signal **Remote Healthy** aktiviert wird. Um zu ermitteln, warum die TCP-Verbindung geschlossen wurde, prüft der Sicherungs-Shelf Manager sofort den Status des Signals **Remote Healthy**. Falls es noch aktiv ist, wird die Prüfung nach kurzer Zeit wiederholt. Wenn das Signal **Remote Healthy** dann inaktiv ist, schließt der Sicherungs-Shelf Manager daraus, dass der aktive Shelf Manager nicht mehr verfügbar ist, und leitet den Switchover (Umschaltung) ein.

Wenn das Signal **Remote Healthy** aktiv bleibt, schließt der Sicherungs-Shelf Manager daraus, dass der Kommunikationslink zwischen den Shelf Manager-Einheiten unterbrochen wurde. In diesem Fall wird kein Switchover (Umschaltung) durchgeführt. Stattdessen initialisiert sich der Sicherungs-Shelf Manager ständig neu und versucht, eine Verbindung zum aktiven Shelf Manager herzustellen, bis der Kommunikationslink wiederhergestellt ist. Die Neuinitialisierung erfolgt, indem die Shelf-Verwaltungskarte neu gestartet und nach dem Neustart auch der Shelf Manager neu gestartet wird. Eine spezielle Logik im Shelf Manager sorgt dafür, dass er nicht versucht, beim Starten zur aktiven Einheit zu werden, wenn der gleichgeordnete Shelf Manager bereits aktiv ist.

Der Shelf Manager verwendet einen Watchdog-Timer, um sich davor zu schützen, dass er aufgrund von Endlosschleifen oder anderen Softwarefehlern nicht mehr reagiert. Wenn der Watchdog-Timer für den aktiven Shelf Manager ausgelöst wird, wird die entsprechende Shelf-Verwaltungskarte zurückgesetzt. Daraufhin wird das Signal **Remote Healthy** für die Sicherungs-Shelf-Verwaltungskarte deaktiviert und auf diese Weise ein Switchover (Umschaltung) initiiert.

Nach einem Switchover (Umschaltung) wird der jetzt aktive Shelf Manager neu initialisiert, die zwischengespeicherten Statusinformationen werden aktiviert und die erforderlichen Informationen von den IPM-Controllern des IPMB werden erfasst. Der aktive Shelf Manager legt das ShMC-Gerät (Adresse 20h) auf dem IPMB offen und übernimmt die IP-Adresse, die für das RMCP und andere externe Interaktionen zwischen dem zuvor aktiven Shelf Manager und dem System Manager verwendet wurde. Da die RMCP-Sitzungsinformationen vom aktiven Shelf Manager auf den Sicherungs-Shelf Manager übertragen werden, „überleben" die RMCP-Sitzungen den Switchover (Umschaltung). Für den System Manager ist der Switchover (Umschaltung) bei Verwendung von RMCP transparent.

Nach dem Switchover (Umschaltung) kann der zuvor aktive Shelf Manager sich deaktivieren oder sich als neuer Sicherungs-Shelf Manager neu initialisieren. Die Initialisierung als Sicherungs-Shelf Manager erfordert einen Neustart des Betriebssystems auf der zuvor aktiven Shelf-Verwaltungskarte.

# Optionen der Systemadministratorschnittstelle

Ein weiteres wichtiges Untersystem des Shelf Managers implementiert die Systemadministratorschnittstelle. Der Systemadministrator ist ein logischer Begriff, der sowohl für Software als auch für menschliche Bediener in einer Leitstelle stehen kann. Der Shelf Manager stellt zwei Systemadministratorschnittstellen-Optionen bereit, die verschiedene Zugriffsmechanismen auf ähnliche Arten von Shelf-Informationen und -Steueroptionen ermöglichen:

- IPMI Local Area Network-Schnittstelle (LAN)
- Command-Line Interface (CLI)

Die IPMI-LAN-Schnittstelle wird verwendet, um die Interoperabilität zwischen unabhängig voneinander implementierten Shelf-Produkten zu erhöhen. Diese Schnittstelle ist in der ATCA-Spezifikation als erforderlich angegeben und unterstützt IPMI-Messaging mit dem Shelf Manager per RMCP. Ein Systemadministrator, der zum Kommunizieren mit Shelves RMCP verwendet, kann in der Regel auch mit anderen ATCA-kompatiblen Shelf Manager-Einheiten kommunizieren. Diese Low-Level-Schnittstelle ermöglicht den Zugriff auf die IPMI-Funktionen eines Shelfs, indem der Shelf Manager als Proxy verwendet wird. Dies umfasst auch das Ausgeben von IPMI-Befehlen an IPM-Controller auf Shelf-Ebene durch den Systemadministrator.

Bei RMCP handelt es sich um eine Standardnetzwerkschnittstelle zu einem IPMI-Controller über ein LAN, die in der IPMI 1.5-Spezifikation definiert ist.

Das CLI umfasst eine umfangreiche Reihe von Textbefehlen, die für den Shelf Manager entweder über eine physikalische serielle Verbindung oder über eine Telnet-Verbindung ausgegeben werden können.

# Systemadministrationsaufgaben

Die Systemadministration für Netra CT 900 Server umfasst normalerweise Installations-, Konfigurations- und Administrationsaufgaben.

Die Solaris OS-Administration auf dem Netra CT 900 Server (einschließlich des Hinzufügens von Solaris-Benutzerkonten) wird durchgeführt, indem eine Anmeldung beim Node Board erfolgt. Die Administration für den Netra CT 900 Server wird durchgeführt, indem eine Anmeldung bei der Shelf-Verwaltungskarte

erfolgt und das CLI der Shelf-Verwaltungskarte verwendet wird. Sie können die Shelf-Verwaltungskarte beim Netra CT 900 Server als zentralen Punkt für die Konfiguration und Administration nutzen.

Die Systemadministrationsaufgaben sind in den folgenden Kapiteln beschrieben.

# Zuordnung von physikalischen Adressen zu logischen Steckplätzen

Wenn Sie auf die Vorderseite des Netra CT 900 Servers sehen, sind die physikalischen Steckplätze von links nach rechts aufsteigend nummeriert. In TABELLE 1-4 ist die Zuordnung der physikalischen Steckplätze zu den logischen Steckplätzen mit den dazugehörigen Adressen aufgeführt.

**TABELLE 1-4** Zuordnung von physikalischen Adressen zu logischen Steckplätzen

| Physikalischer Steckplatz | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | ShMM #1 | ShMM #2 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Logischer Steckplatz | 13 | 11 | 9 | 7 | 5 | 3 | 1 | 2 | 4 | 6 | 8 | 10 | 12 | 14 | N/V | N/V |
| HW-Adresse (hex) | 4D | 4B | 49 | 47 | 45 | 43 | 41 | 42 | 44 | 46 | 48 | 4A | 4C | 4E | 8 | 9 |
| IPMB-Adresse (hex) | 9A | 96 | 92 | 8E | 8A | 86 | 82 | 84 | 88 | 8C | 90 | 94 | 98 | 9C | 10 | 12 |

KAPITEL **2**

# Konfigurieren des Systems

In diesem Kapitel wird vorausgesetzt, dass Sie das Solaris-Betriebssystem und die erforderlichen Patches bereits auf dem Netra CT 900 Server und den Netra ATCA Node Boards installiert haben.

Das Netra CT 900-System wird hauptsächlich über das Command-Line Interface (CLI) der aktiven Shelf-Verwaltungskarte konfiguriert. Das CLI der aktiven Shelf-Verwaltungskarte ermöglicht die Konfiguration, Administration und Verwaltung auf Systemebene. Dies umfasst die Node Boards, die Switching Fabric Boards, die Shelf-Verwaltungskarte, die Stromversorgungsmodule und die Lüfterelemente. Das CLI der Shelf-Verwaltungskarte kann sowohl lokal als auch remote verwendet werden.

Dieses Kapitel enthält folgende Abschnitte:



## Zugreifen auf die Shelf-Verwaltungskarten

Der Shelf Manager wird basierend auf einer speziellen Linux-Implementierung ausgeführt. Die unterste Linux-Schicht ist der U-Boot Firmware Monitor. Wenn Sie zum ersten Mal auf eine der Shelf-Verwaltungskarte (ShMM) zugreifen, müssen Sie diesen Schritt über den seriellen Port (Konsole) durchführen, indem Sie ein ASCII-Terminal oder das TIP-Programm verwenden. Jede Shelf-Verwaltungskarte

unterstützt mehrere gleichzeitige Sitzungen (TIP- und Telnet-Verbindungen). Die TCP/IP-Standardadresse für die aktive Shelf-Verwaltungskarte lautet 192.168.0.2.

Wenn Sie die Verbindung zu einer Shelf-Verwaltungskarte über einen seriellen Port herstellen, verbinden Sie ein serielles Terminal oder einen Emulator mit einem der beiden seriellen Ports an der Vorderseite des Shelf-Alarmbedienfelds (Shelf Alarm Panel, SAP). Verwenden Sie den seriellen Port 1, um die Verbindung zur oberen Shelf-Verwaltungskarte (ShMM1) herzustellen. Dies ist standardmäßig die *aktive* Shelf-Verwaltungskarte. Über den seriellen Port 2 wird die Verbindung zur standardmäßigen *Sicherungs*-Karte (ShMM2) hergestellt. ABBILDUNG 2-1 zeigt die Anordnung der seriellen Ports für die obere und die untere Shelf-Verwaltungskarte. Die Terminal- bzw. Modemeinstellung muss auf 115200, N, 8, 1 eingestellt sein.

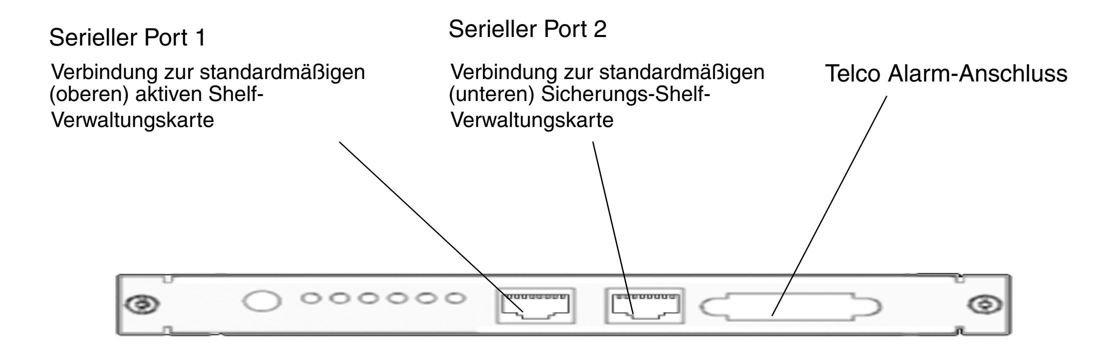



**ABBILDUNG 2-1** Anschlüsse des Shelf-Alarmbedienfelds

Wenn Sie zum ersten Mal auf die Shelf-Verwaltungskarte zugreifen, melden Sie sich als root an und verwenden das Standardpasswort sunct900. Dieses Konto ist mit allen Rechten (Berechtigungen) ausgestattet. Das Konto kann außerdem nicht gelöscht werden. Sie sollten das Passwort für dieses Konto jedoch aus Sicherheitsgründen ändern, bevor der Netra CT 900 Server in Betrieb geht.

Verwenden Sie den Linux-Befehl `passwd` wie folgt, um das root-Passwort zu ändern:

```
# passwd

Changing password for root
Enter the new password (minimum of 5, maximum of 8 characters)
Please use a combination of upper and lower case letters and
numbers.
Enter new password: xxxxxxx
Re-enter new password: xxxxxxx
Password changed.
#
```

Verwenden Sie nach dem Anmelden den Befehl `clia shmstatus`, um sicherzustellen, dass Sie bei der *aktiven* Shelf-Verwaltungskarte angemeldet sind, bevor Sie fortfahren. Wenn Sie bei der Standby-Shelf-Verwaltungskarte angemeldet sind, können Sie den Befehl `clia switchover` verwenden, um die Shelf-Verwaltungskarte zur `aktiven` Karte zu machen, oder melden Sie sich ab, und melden Sie sich bei der aktiven Shelf-Verwaltungskarte an. (Weitere Informationen finden Sie unter „`shmstatus`“ auf Seite 264 und „`switchover`“ auf Seite 266.)

Die folgenden Abschnitte enthalten Informationen zur Konfiguration der Ethernet-Ports der Shelf-Verwaltungskarte und zur Einrichtung von Benutzerkonten und Passwörtern mithilfe des Shelf-Verwaltungskarte-CLI. Weitere Informationen zur Verwendung des Shelf-Verwaltungskarte-CLI finden Sie unter Kapitel 3.

---

**Hinweis –** Der Ausdruck *Shelf-Verwaltungskarte* wird in diesem Handbuch entweder für die aktive oder die Standby-Shelf-Verwaltungskarte verwendet, falls keine anderslautenden Angaben gemacht werden. In diesem Handbuch wird für beide Karten auch die Kurzform `ShMM #` verwendet.

---

# Einrichten von U-Boot

Beim Hochfahren und Neustarten der Shelf-Verwaltungskarte (ShMM) beginnt die Hardware mit dem Ausführen der U-Boot-Firmware im Flash-Speicher. Die Firmware führt die ShMM-Basisinitialisierung durch und beginnt mit dem Starten des Linux-Kernels, es sei denn, der Benutzer deaktiviert die Autoboot-Funktion explizit (und zwingt die Firmware so dazu, zur Benutzerbefehlsschnittstelle für die Wartung umzuschalten). Linux wird über den Kernel und über `root`-Dateisystemabbilder (Images) im Flash-Speicher gestartet. U-Boot fügt das Kernel-Abbild in den RAM ein, richtet Kernel-Parameter ein und übergibt die Steuerung an die Kernel-Einsprungstelle.

## U-Boot-Oberfläche

U-Boot ist über den seriellen Port des ShMM zugänglich und erfordert eine Konfiguration, die auf die verwendete Betriebsumgebung abgestimmt ist. Wenn das ShMM hochgefahren ist, werden die folgenden Informationen auf der Konsole angezeigt:

```
U-Boot 1.1.2 (Nov 11 2005 - 11:32:08)

CPU: Au1550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 08004610
DRAM:  128 MB
Flash:  64 MB
In:    serial
Out:   serial
Err:   serial
Net:   Au1X00 ETHERNET
Hit any key to stop autoboot:  0
#
```

# ist die Eingabeaufforderung, an der Benutzer Befehle eingeben können.

# U-Boot-Umgebungsvariablen

U-Boot enthält eine Reihe von Umgebungsvariablen, die Sie vor der Verwendung konfigurieren müssen. In TABELLE 2-1 sind die verfügbaren Standardvariablen beschrieben.

**TABELLE 2-1** Standardmäßige U-Boot-Umgebungsvariablen

| **Umgebungsvariable** | **Beschreibung** |
|---|---|
| addmisc | Hängt die Einstellungen quiet, reliable_upgrade und console an bootargs an. Diese Variable wird in der Regel nicht geändert. |
| baudrate | Baudrate des seriellen Ports, standardmäßig 115200. |
| bootargs | Befehlszeile, die an den Linux-Kernel übergeben wird. Kann Verweise auf andere U-Boot-Umgebungsvariablen enthalten, die zur Laufzeit aufgelöst werden. Der Standardwert lautet:<br>root=/dev/ram rw console=ttyS0,115200 reliable_upgrade=y |
| bootcmd | U-Boot-Befehl, der ausgeführt wird, um den automatischen Startvorgang (Autoboot) durchzuführen. |
| bootdelay | Autoboot-Verzögerungswert in Sekunden. |
| bootfile | Parameter, der angibt, welches Kernel-Abbild von den Startoptionen net und nfs verwendet wird. |
| console | Einstellung für den Kernel und für den Port und die Baudrate der init-Skriptkonsole. Die Standardeinstellung lautet console=ttyS0,115200. |
| ethaddr | MAC-Adresse des primären On-Chip-Ethernet-Controllers. Der Wert dieser Variablen wird automatisch von U-Boot eingestellt. Diese Adresse wird an den Kernel-Ethernet-Treiber übergeben. |
| eth1addr | MAC-Adresse des sekundären Ethernet-Controllers. Der Wert dieser Variablen wird automatisch von U-Boot eingestellt. Diese Adresse wird an den Kernel-Ethernet-Treiber übergeben. |
| flash_reset | Weist Linux an, die Flash-Dateisysteme zu löschen (/etc und /var) und die Werkseinstellungen wiederherzustellen (y/n). Das Startskript des Systems setzt diese Variable nach dem Löschen des Flash-Speichers zurück auf n. Die Standardeinstellung lautet n. |
| gateway | Standard-IP-Adresse für das Gateway. Diese Variable kann als Teil einer Kernel-Befehlszeile übergeben werden, um die Weiterleitungseinstellungen für Netzwerkschnittstellen automatisch zu konfigurieren. |
| hostname | Netzwerkhostname, Standardeinstellung lautet sentry. |

**TABELLE 2-1** Standardmäßige U-Boot-Umgebungsvariablen *(Fortsetzung)*

| Umgebungsvariable | Beschreibung |
|---|---|
| `io_config` | Bestimmt, ob die PSC-Controller für die Dual-Slave-Adressenkonfiguration eingerichtet wurden (`y`/`n`). Standardeinstellung: `y`. |
| `ipaddr` | IP-Adresse, die von der primären On-Chip-Ethernet-Schnittstelle verwendet wird. Diese Variable wird verwendet, um die durch `ipdevice` angegebene Netzwerkschnittstelle automatisch zu konfigurieren, wenn die Variable `rc_ifconfig` auf `y` gesetzt ist. Beachten Sie, dass das Startskript des Systems das am wenigsten signifikante Bit der Hardwareadresse für den ShMM-Carrier einstellt. Wenn die Hardwareadresse also ein gerader Wert ist, wird das letzte Bit der IP-Adresse auf `0` gesetzt. Anderenfalls wird es auf `1` gesetzt. Dies erfolgt im Startskript `/etc/netconfig`, um koordinierte IP-Adressenkonfigurationen für redundante ShMMs zu unterstützen. Wenn Sie diese Funktion deaktivieren möchten, entfernen Sie einfach die Datei `/etc/readhwaddr`. |
| `ip1addr` | IP-Adresse, die von der sekundären Ethernet-Schnittstelle verwendet wird. Diese Variable kann als Teil einer Kernel-Befehlszeile übergeben werden, um die entsprechende Kernel-Netzwerkschnittstelle automatisch zu konfigurieren. |
| `ipdevice` | Das entsprechende Gerät zu `ipaddr`, Standardeinstellung lautet `eth0`. |
| `ip1device` | Das entsprechende Gerät zu `ip1addr`, Standardeinstellung lautet `eth1`. |
| `kernel_start` | Die absolute Startadresse des Kernel-Abbilds im Flash-Speicher. Diese Variable wird von U-Boot während des Urladens (Bootstrapping) automatisch eingestellt. |
| `logging` | Gibt an, ob die Meldungsprotokolldatei im `ram` oder im `flash` vorgehalten werden soll. Die Standardeinstellung lautet `ram`. Dies ist auch die empfohlene Option. |
| `net` | Sie können diese Variable als Ersatz für `bootcmd` verwenden, um einen Kernel und ein RFS-Abbild von einem TFTP-Server zu starten. |
| `netmask` | Netzwerkmaske, Standardeinstellung ist `255.255.255.0`. |
| `password_reset` | Weist Linux an, das werkseitig vorgegebene Standardpasswort wiederherzustellen (also für den Benutzer `root` ein leeres Passwortfeld). Die Standardeinstellung lautet `n`. |
| `post_normal` | Bestimmt die Liste der POST-Tests, die bei jedem Startvorgang durchgeführt werden. Wenn keine Einstellung vorhanden ist, werden die bei der Kompilierung vorhandenen Standardeinstellungen verwendet. Die Namen der Tests, die in einem Wert dieser Variable aufgeführt sind, sind durch Leerstellen voneinander getrennt. |

**TABELLE 2-1** Standardmäßige U-Boot-Umgebungsvariablen *(Fortsetzung)*

| Umgebungsvariable | Beschreibung |
|---|---|
| `post_poweron` | Bestimmt die Liste der POST-Tests, die nur nach einem Kaltstart durchgeführt werden (anstatt bei jedem Startvorgang). Wenn keine Einstellung vorhanden ist, werden die bei der Kompilierung vorhandenen Standardeinstellungen verwendet. Die Namen der Tests, die in einem Wert dieser Variable aufgeführt sind, werden durch Leerstellen voneinander getrennt. |
| `quiet` | Weist den Kernel beim Starten an, Fortschrittsmeldungen nicht auf der seriellen Konsole anzuzeigen. Die Standardeinstellung lautet `quiet=quiet`. |
| `ramargs` | Stellt die Kernel-Befehlszeile in der Variablen `bootargs` je nach `root`-Dateisystem ein, das über eine „ramdisk“ gemountet werden soll. |
| `ramdisk` | Gibt an, welches RFS-Abbild von den Startoptionen `net` und `nfs` verwendet wird. |
| `ramsize` | Größe des Systemspeichers in Byte. Standardeinstellung: Wird aus der SDRAM-Konfigurationscodierung des Erstellungszeit-Konfigurationsblocks berechnet. |
| `rc_ifconfig` | Ermöglicht es dem Skript `/etc/rc`, die IP-Adresse einzurichten, anstatt sie aus der Datei `shelfman` abzurufen. Die Standardeinstellung lautet `n` (`shelfman` kann IP-Adressen einrichten). |
| `rc2` | Gibt das sekundäre RC-Skript an, das aufgerufen werden soll. Dies ist das trägerspezifische Startskript. Die Standardeinstellung lautet `/etc/rc.acb3` (bzw. ein anderes Skript, das für die jeweilige Zielplattform geeignet ist). |
| `reliable_upgrade` | Bestimmt, ob auf dem ShMM-500 die zuverlässige Prozedur für Softwareaktualisierungen aktiviert ist (`y`/`n`). Standardeinstellung: `y`. Das Setzen dieser Variablen auf `n` wird momentan nicht unterstützt. Wenn Sie die Variable auf `n` setzen, wird beim nächsten Starten des ShMM eine Fehlermeldung ausgegeben, und die Karte hängt. |
| `rfs_start` | Die absolute Startadresse des `root`-Dateisystemabbilds im Flash-Speicher. Diese Variable wird von U-Boot während des Urladens (Bootstrapping) automatisch eingestellt. |
| `rmcpaddr` | Standardmäßige IP-Adresse für den RMCP-Dienst. |
| `serverip` | IP-Adresse des TFTP-Servers. |

**TABELLE 2-1** Standardmäßige U-Boot-Umgebungsvariablen *(Fortsetzung)*

| Umgebungsvariable | Beschreibung |
|---|---|
| `start_rc2_daemons` | Weist das sekundäre Startskript an, nach dem Starten die Dämonen `snmpd/boa` und `shelfman` zu starten bzw. nicht zu starten. Die Standardeinstellung lautet `y`. |
| `time_server` | Zeitserver für die Synchronisierung zur Laufzeit. Wenn diese Variable nicht angegeben ist, wird von der Hardwareuhr beim Systemstart Zeit abgezogen. HINWEIS: Wenn diese Variable angegeben ist, muss die Variable `ip1device` auf `usb0` gesetzt sein, um eine ordnungsgemäße Synchronisierung zu ermöglichen. |
| `timezone` | Lokale Zeitzone im Format „CCC*n*", wobei *n* die Differenz zur Greenwich Mean Time (GMT) angibt und somit auch negativ sein kann. CCC steht für die Zeitzone. Die Standardeinstellung lautet `UTC0`. |

# Zuweisen von Werten zu Umgebungsvariablen

Verwenden Sie das folgende Format, um einer Umgebungsvariablen einen Wert zuzuweisen:

`setenv` *variablenname neuer_wert*

Beispiel:

```
# setenv bootdelay 1
```

Nachdem Sie alle Umgebungsvariablen richtig eingestellt haben, müssen Sie diese in den Flash-Speicher zurückspeichern, damit sie auch verfügbar sind, wenn das ShMM heruntergefahren wird. Hierfür wird der Befehl `saveenv` verwendet.

```
# saveenv
```

Die Funktion `setenv` ist auch als Linux-Dienstprogramm verfügbar, das auf die gleiche Weise verwendet werden kann. Verwenden Sie das zusätzliche Dienstprogramm `getenv`, um U-Boot-Variablen an der Shell-Eingabeaufforderung anzuzeigen.

# Umgebungsvariablen zur Konfiguration für den Shelf Manager

Wenn Sie U-Boot zum ersten Mal starten, werden die folgenden standardmäßigen Umgebungsvariablen definiert:

```
bootcmd=run setup_bootargs; bootm BFB00000 BFC40000
bootdelay=3
baudrate=115200
ethaddr= 00:00:1a:18:xx:yy
eth1addr= 00:00:1a:18:xx:zz
serverip=192.168.0.7
netmask=255.255.0.0
hostname=sentry
gateway=192.168.0.1
ipdevice=eth0
ip1addr=192.168.1.3
ip1device=eth1
rc2=/etc/rc.acb3
ipaddr=192.168.0.2
start_rc2_daemons=y
flash_reset=n
password_reset=n
logging=ram
rc_ifconfig=n
bootfile=sentry.mips.kernel
ramdisk=sentry.mips.rfsnet=tftp 80400000 $(bootfile); tftp
80800000 $(ramdisk); bootm 80400000 80800000
rmcpaddr=192.168.1.15
timezone=EST
bootargs=root=/dev/ram rw console=ttyS0,115200 reliable_upgrade=y
```

Mehrere dieser Umgebungsvariablen müssen mit Werten neu konfiguriert werden, die sich für den Netzwerkkontext eignen, in dem die ShMMs verwendet werden sollen.

# Konfigurieren von Ethernet-Ports für Shelf-Verwaltungskarten

Jede Shelf-Verwaltungskarte verwendet zwei Ethernet-Ports, die mit den redundanten Switch-Karten verbunden sind. Da ATCA nur RMCP als shelfexterne Schnittstelle benötigt, wird auf die externen Ethernet-Ports des Shelfs als RMCP-Ports verwiesen, obwohl die anderen externen Schnittstellen (Telnet) des Shelfs auch über diesen Port verfügbar sind.

Wenn die Verbindung hergestellt ist, müssen Sie bei der Shelf-Verwaltungskarte mit einem Benutzerkonto angemeldet sein, das über alle Berechtigungen verfügt. Sie konfigurieren die Ports mit CLI-Befehlen und starten die Shelf-Verwaltungskarte dann neu, damit die Änderungen wirksam werden.

## Verwenden der ersten Ethernet-Schnittstelle

Da der RMCP-Ethernet-Port direkt mit dem lokalen Netzwerk verbunden ist, müssen Sie die IP-Adresse gemäß dem Netzwerk einrichten. Wenn am jeweiligen Standort z. B. der IP-Adressenbereich `192.168.0.x` verwendet wird, müssen Sie den RMCP-Ethernet-Port auf eine eindeutige IP-Adresse dieses Bereichs einstellen. Beispiel: `192.168.0.2`. Bei einer Einrichtung mit einem redundanten ShMM ist nur für ein ShMM (das aktive ShMM) die RMCP-IP-Adresse des RMCP-Ethernet-Ports aktiviert. Das Sicherungs-ShMM weist dem RMCP-Ethernet-Port dieselbe IP-Adresse zu, die jedoch nur aktiviert wird, wenn dieses ShMM die aktive Rolle übernimmt. Auf diese Weise ist die Verfügbarkeit der RMCP-IP-Adresse in einer Failoversituation sichergestellt.

### Zuweisen einer zusätzlichen IP-Adresse zur ersten Netzwerkschnittstelle

In der Standardkonfiguration ist der ersten Netzwerkschnittstelle keine IP-Adresse zugewiesen (und das ShMM ist nicht über das Netzwerk zugänglich), bis der Shelf Manager gestartet und die RMCP-IP-Adresse zugewiesen wird. In einigen Fällen kann es jedoch hilfreich sein, der RMCP-Netzwerkschnittstelle eine IP-Adresse zuzuweisen, damit das ShMM über das Netzwerk zugänglich ist, sobald das Betriebssystem gestartet wurde. Dabei ist es auch ratsam, die RMCP-IP-Adresse zusätzlich zur ursprünglich zugewiesenen IP-Adresse zu verwenden, anstatt sie beim Starten des Shelf Managers ersetzen zu lassen.

Bei dieser Konfiguration müssen Sie den Shelf Manager anweisen, die RMCP-IP-Adresse nicht dem ersten Netzwerkadapter selbst zuzuweisen (eth0), sondern seinem ersten Alias (eth0:1). Die ursprüngliche IP-Adresse wird dann dem Netzwerkadapter selbst zugewiesen (eth0), wenn das Betriebssystem gestartet wird. Diese ursprüngliche Zuweisung erfolgt im Initialisierungsskript /etc/rc. Es werden die folgenden Schritte ausgeführt:

1. Aktivieren der U-Boot-Variable rc_ifconfig (wie folgt):

```
setenv rc_ifconfig y
```

2. Zuweisen der ursprünglichen IP-Adresse zur U-Boot-Variablen ipaddr. Beispiel:

```
setenv ipaddr 192.168.1.240
```

3. Ändern des Werts für den RMCP_NET_ADAPTER in der Shelf Manager-Konfigurationsdatei /etc/shelfman.conf in eth0:1. Beispiel:

```
RMCP_NET_ADAPTER = eth0:1
```

Bei einer redundanten Konfiguration kann die U-Boot-Variable ipaddr für beide ShMMs denselben Wert haben. Die ursprüngliche IP-Adresse, die den beiden redundanten ShMMs jeweils zugewiesen ist, basiert auf dem Wert von ipaddr. Dieser Wert wird jedoch je nach Hardwareadresse des ShMM geändert. Das am wenigsten signifikante Bit der IP-Adresse wird auf das am wenigsten signifikante Bit der Hardwareadresse gesetzt. Im obigen Beispiel würde die IP-Adresse für das ShMM mit einer gerade Hardwareadresse 192.168.1.240 und für das ShMM mit einer ungeraden Hardwareadresse 192.168.1.241 lauten. Sie können diese Änderung der IP-Adresse deaktivieren, indem Sie die Datei /etc/readhwaddr entfernen.

## Propagieren der RMCP-Adresse

Eine optionale Funktion des Shelf Managers ermöglicht es, dass das Sicherungs-ShMM ebenfalls über das externe Netzwerk zugänglich gemacht wird. Dies erfolgt mithilfe einer IP-Adresse, die sich von der RMCP-IP-Adresse nur durch das am wenigsten signifikante Bit unterscheidet. Die Netzmaske und das Standardgateway sind für das Sicherungs-ShMM identisch mit den Einstellungen für das aktive ShMM. Wenn die RMCP-IP-Adresse z. B. 192.168.0.2 lautet, hat das Sicherungs-ShMM entsprechend die IP-Adresse 192.168.0.3 und verfügt über dieselbe Einstellung für Netzmaske und Standardgateway. Um diese Funktion zu aktivieren,

ist es notwendig, den Shelf Manager-Konfigurationsparameter PROPAGATE_RMCP_ADDRESS in der Shelf Manager-Konfigurationsdatei (/etc/shelfman.conf) auf TRUE zu setzen.

# Verwenden der zweiten Ethernet-Schnittstelle

Die zweite Netzwerkschnittstelle verbindet den Shelf Manager mit einem der Hub Boards des ATCA-Netzwerks. Für die Kommunikation zwischen den redundanten Shelf Manager-Einheiten werden Dual USB-Netzwerkschnittstellen verwendet.

# Verwenden von Dual USB-Netzwerkschnittstellen für die redundante Kommunikation

Auf dem ShMM werden über die beiden USB-Verbindungen zwei weitere Netzwerkschnittstellen implementiert. Bei dieser Konfiguration sorgen sie immer für die Verbindung der beiden redundanten Shelf Manager. Diese Schnittstellen haben die Namen usb0 und usb1. Die Schnittstelle usb0 ist immer vorhanden, während die Schnittstelle usb1 nur vorhanden ist, wenn die Schnittstelle usb0 des gleichgeordneten Shelf Managers aktiv ist (also wenn der gleichgeordnete Shelf Manager physikalisch installiert ist und ausgeführt wird). Die Schnittstellen verfügen außerdem über eine Überkreuzverbindung: usb0 des ersten Shelf Managers ist mit usb1 des zweiten Shelf Managers verbunden (und umgekehrt).

Der Shelf Manager unterstützt den Einsatz von USB-Netzwerkschnittstellen für die Kommunikation zwischen den redundanten Shelf Managern. Wenn Sie diese Funktion nutzen möchten, müssen Sie in der Shelf Manager-Konfigurationsdatei /etc/shelfman.conf wie folgt zwei Redundanz-Netzwerkadapter definieren:

```
REDUNDANCY_NET_ADAPTER = "usb0"
REDUNDANCY_NET_ADAPTER2 = "usb1"
```

Ein weiterer Aspekt betrifft die Definition der Teilnetzmaske für die Redundanz-Netzwerkschnittstellen. Im Legacy-Fall, in dem nur ein redundanter Netzwerkadapter verwendet wird, werden aus der Redundanz-IP-Adresse, die unter /etc/shelfman.conf angegeben ist, zwei IP-Adressen abgeleitet. Sie werden den beiden Endpunkten der Redundanzverbindung zugewiesen und unterscheiden sich nur hinsichtlich des letzten signifikanten Bits.

Wenn jedoch zwei Redundanz-Netzwerkadapter genutzt werden, werden vier verschiedene IP-Adressen verwendet, und zwar eine für jeden Endpunkt (jeweils zwei Endpunkte für die beiden redundanten Shelf Manager). Um die ordnungsgemäße Funktionsweise sicherzustellen, müssen beide Endpunkte eines Shelf Managers (usb0 und usb1) verschiedenen logischen Netzwerken angehören.

usb0 auf dem einen Shelf Manager und usb1 auf dem anderen Shelf Manager müssen jedoch demselben logischen Netzwerk angehören. Basierend auf diesen Aspekten werden die zwei zusätzlichen IP-Adressen abgeleitet, indem jeweils das am wenigsten signifikante Bit der Teilnetzmaske der Redundanz-IP-Adresse geändert wird, die unter /etc/shelfman.conf angegeben ist. Aus diesem Grund gelten für die Teilnetzmaske strengere Maßstäbe als für die Standardeinstellung der angegebenen Redundanz-IP-Adressenklasse. Wenn keine Teilnetzmaske angegeben wird, wird sie standardmäßig auf 255.255.255.128 gesetzt. Dies ist auch der empfohlene Wert für diesen Parameter in /etc/shelfman.conf, wenn zur Sicherstellung der Redundanz USB-Netzwerkschnittstellen verwendet werden.

Das folgende Beispiel zeigt, wie IP-Adressen für die USB-Netzwerkschnittstellen abgeleitet werden.

Angenommen, in /etc/shelfman.conf liegen die folgenden Definitionen vor:

```
REDUNDANCY_IP_ADDRESS = 192.168.1.2
REDUNDANCY_NETMASK = 255.255.255.128
```

In diesem Fall verfügt das ShMM mit der *geraden* Hardwareadresse über die folgende IP-Adressenzuweisung:

usb0: 192.168.1.2 (keine Änderungen)
usb1: 192.168.1.130 (das am wenigsten signifikante Bit der Netzmaske wird geändert)

Das ShMM mit der *ungeraden* Hardwareadresse verfügt über die folgende IP-Adressenzuweisung:

usb0: 192.168.1.131 (das am wenigsten signifikante Bit der IP-Adresse und das am wenigsten signifikante Bit der Netzmaske wird geändert)

usb1: 192.168.1.3 (das am wenigsten signifikante Bit der IP-Adresse wird geändert)

# Ändern der standardmäßigen ShMM-Netzwerkparameter

Die Konfiguration eines ShMM für den Betrieb in einer speziellen Netzwerkumgebung erfordert das Ändern der folgenden Netzwerkparameter:

- RMCP-IP-Adresse
- RMCP-GATEWAY-Adresse
- RMCP-Netzmaske

Das Ändern der RMCP-Netzwerkparameter erfolgt in mehreren Schritten. Zuerst müssen Sie die Umgebungsvariablen des U-Boot-Netzwerks aktualisieren, und dann müssen Sie die Netzwerkeinstellungen des gestarteten AKTIVEN ShMM aktualisieren, indem Sie das Shelf Manager-CLI verwenden.

## ▼ So ändern Sie die standardmäßigen ShMM-Netzwerkparameter

1. **Schließen Sie eine Konsolenverbindung für den seriellen Port an das ShMM an.**

   Der Wert hierfür lautet normalerweise 115.200 Baud, N/8/1. Starten Sie den ShMM-Träger neu, und drücken Sie die Leertaste, um den automatischen Startvorgang zu unterbrechen. Folgendes wird angezeigt:

```
U-Boot 1.1.2 (Nov 11 2005 - 11:32:08)

CPU: Au1550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 00 00 00 00 00 00 00 00 00 03 03 03
DRAM:  128 MB
Flash: 64 MB
In:    serial
Out:   serial
Err:   serial
Net:   Au1X00 ETHERNET
Hit any key to stop autoboot:  0
ShMM #
```

2. **Wiederholen Sie die aktuellen Netzwerkeinstellungen.**

```
ShMM # printenv rmcpaddr netmask gateway
rmcpaddr=192.168.0.44
netmask=255.255.255.0
gateway=192.168.0.1
ShMM #
```

3. **Ändern Sie die Einstellungen, und legen Sie diese im nicht-flüchtigen Speicher ab.**

```
ShMM # setenv rmcpaddr 10.1.1.10
ShMM # setenv netmask 255.255.0.0
ShMM # setenv gateway 10.1.1.1
ShMM # saveenv
Un-Protected 1 sectors
Erasing sector  0 ... Erasing sector at 0x  800000
ok.
Saving Environment to EEPROM...done.
ShMM #
```

4. **Starten Sie das ShMM wie für den normalen Betrieb, und melden Sie sich als Benutzer `root` an.**

```
ShMM # reset

U-Boot 1.1.2 (Nov 11 2005 - 11:32:08)

CPU: Au1550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 00 00 00 00 00 00 00 00 00 03 03 03
DRAM:  128 MB
Flash: 64 MB
In:    serial
Out:   serial
Err:   serial
Net:   Au1X00 ETHERNET
Hit any key to stop autoboot:  0
## Booting image at bfb00000 ...
   Image Name:   MIPS Linux-2.4.26
   Created:      2005-05-07  17:35:21 UTC
   Image Type:   MIPS Linux Kernel Image (gzip compressed)
   Data Size:    843144 Bytes = 823.4 kB
   Load Address: 80100000
   Entry Point:  802bc040
   Verifying Checksum ... OK
   Uncompressing Kernel Image ... OK
## Loading Ramdisk Image at bfc40000 ...
   Image Name:   sentry RFS Ramdisk Image
…
…
sentry login: root

BusyBox v0.60.5 (2005.05.07-17:27+0000) Built-in shell (msh)
#
```

5. **Lassen Sie das Starten des ShMM zu.**

---

**Hinweis –** Die Einstellungen, die in der U-Boot-Firmware geändert wurden, werden nicht automatisch in die Linux-Umgebung propagiert. Der Grund hierfür besteht darin, dass der Shelf Manager seine eigene Kopie der Netzwerkkonfigurationsdaten beibehalten muss, um Failoversituationen verwalten zu können.

---

Wenn dies das erste Mal ist, dass der Shelf Manager gestartet wird, oder wenn die Flash-Geräte vor dem Starten auf die Werkseinstellungen zurückgesetzt wurden, verwendet der Shelf Manager die von U-Boot bereitgestellten Netzwerkeinstellungen, um diesen Netzwerkkontext einzurichten (dies bedeutet, dass die Änderungen, die Sie in U-Boot vorgenommen haben, propagiert werden).

Anderenfalls sind die folgenden Schritte nötig, um die Netzwerkeinstellungen im Shelf Manager-Kontext zu konfigurieren.

6. **Überprüfen Sie, ob Sie der aktive Shelf Manager sind.**

   Sie müssen Änderungen nur auf der aktiven Shelf-Verwaltungskarte vornehmen, indem Sie den Befehl `cpld` verwenden. Dieser Befehl aktualisiert über die Redundanz-Schnittstelle auch das Sicherungsmodul mit den Netzwerkkonfigurationseinstellungen. Wenn Sie nicht das aktive ShMM sind, stellen Sie eine Verbindung mit dem anderen ShMM-Gerät her und fahren mit Schritt 7 fort.

```
# cpld
CPLD word: E806
        0002h - Local Healthy
        0004h - Switchover Request Local
        0800h - Hot Swap Latch Open
        2000h - Active
        4000h - Interrupt Status
        8000h - Reboot Was Caused By Watchdog
#
```

7. **Rufen Sie die aktuellen IP-Einstellungen ab.**

```
# clia getlanconfig 1

IPM Sentry Shelf Manager Command Line Interpreter

Authentication Type Support: 0x15 ( None MD5 Straight Password/Key )
Authentication Type Enables:
    Callback level: 0x00
    User level: 0x15 ( "None" "MD5" "Straight Password/Key" )
    Operator level: 0x15 ( "None" "MD5" "Straight Password/Key" )
    Administrator level: 0x15 ( "None" "MD5" "Straight Password/Key" )
    OEM level: 0x00
IP Address: 206.25.139.28
IP Address Source: Static Address (Manually Configured) (0x01)
MAC Address: 00:50:c2:22:50:30
Subnet Mask: 0.0.0.0
IPv4 Header Parameters: 0x40:0x40:0x10
Primary RMCP Port Number: 0x026f
Secondary RMCP Port Number: 0x0298
BMC-generated ARP Control: 0x02
    Enable BMC-generated ARP Response
Gratuitous ARP Interval: 2.0 seconds
Default Gateway Address: 206.25.139.3
Default Gateway MAC Address: 00:00:00:00:00:00
Backup Gateway Address: 0.0.0.0
Backup Gateway MAC Address: N/A
Community String: "public"
Number of Destinations: 16
Destination Type:
    N/A
Destination Address:
    N/A
#
```

8. **Ändern Sie die IP-Einstellungen wie unten gezeigt.**

```
# clia setlanconfig 1 ip 10.1.1.10

IPM Sentry Shelf Manager Command Line Interpreter
IP set successfully

# clia setlanconfig 1 subnet_mask 255.255.0.0

IPM Sentry Shelf Manager Command Line Interpreter
Subnet Mask set successfully

# clia setlanconfig 1 dft_gw_ip 10.1.1.1

IPM Sentry Shelf Manager Command Line Interpreter
Default Gateway Address set successfully
#
```

# Einrichten der Shelf Manager-Konfigurationsdatei

Die Shelf Manager-Konfigurationsdatei (`shelfman.conf`) befindet sich im Verzeichnis `/etc`. Jede Zeile der Datei ist entweder eine Kommentarzeile (beginnt mit `#`) oder ein *name = value*-Paar, mit dem die Zuweisung für die Konfigurationsparameter erfolgt. Der *name* (Name) und der *value* (Wert) sind durch ein Gleichheitszeichen (=) getrennt.

Beim Konfigurationsparameternamen wird die Groß-/Kleinschreibung beachtet. Es gibt Konfigurationsparameter der folgenden Typen: boolescher Wert, Zahl, Zeichenfolge oder IP-Adresse.

Das Format des Werts entspricht wie folgt dem Typ des Konfigurationsparameters:

| | |
|---|---|
| Boolescher Wert | Ein boolescher Wert kann entweder durch die Zeichenfolgen `FALSE` oder `TRUE` oder durch die entsprechenden numerischen Werte `0` bzw. `1` ausgedrückt werden. |
| Zahl | Ein ganzzahliger numerischer Wert (ggf. mit Vorzeichen). Die Hexadezimalnotation `0x`... wird ebenfalls unterstützt. |
| Zeichenfolge | Eine Zeichenfolge mit oder ohne Anführungszeichen (doppelte Anführungszeichen " "). In Anführungszeichen gesetzte Zeichenfolgen können Leerstellen enthalten, während nicht in Anführungszeichen gesetzte Zeichenfolgen an der ersten Leerstelle enden. Die maximale Zeichenfolgenlänge ist für jeden Konfigurationsparameter vom Typ Zeichenfolge separat festgelegt. |
| IP-Adresse | Die IP-Adresse wird in der sog. Dezimalpunktnotation angegeben (*xxx.xxx.xxx.xxx*). |

Sie können einen Wert einer Umgebungsvariablen als Konfigurationsparameterwert angeben, indem Sie die Notation `$envvar` verwenden. In diesem Fall wird der Wert der Variablen `envvar` ausgetauscht, wenn die Konfigurationsdatei gelesen wird. Beispiel:

```
DEFAULT_RMCP_IP_ADDRESS = $IPADDR
```

Nachdem der Shelf Manager zum ersten Mal gestartet wurde, werden die IP-Adressen zusammen mit den IPMI-LAN-Konfigurationsparametern gespeichert. Sie können auf die LAN-Konfigurationsparameter zugreifen bzw. diese ändern, indem Sie eine beliebige shelfexterne Schnittstelle (RMCP oder CLI) verwenden. Diese Konfigurationsparameter haben Vorrang vor der Konfigurationsdatei `shelfman`, wenn der Shelf Manager neu gestartet wird. Auf diese Weise soll die Beibehaltung aller Änderungen sichergestellt werden, die an den LAN-IP-Adressen und am Gateway über diese Schnittstellen vorgenommen werden. Wenn der Shelf Manager-IP-Verbindungsdatensatz in den Shelf-FRU-Informationen jedoch eine IP-Adresse enthält, hat diese Vorrang vor allen anderen Einstellungen der shelfexternen bzw. RMCP-IP-Adresse. Es wird empfohlen, diese Adresse entweder in den Shelf-FRU-Informationen nicht anzugeben oder sie auf `0.0.0.0` zu setzen, um sicherzustellen, dass Adressen über die Shelf Manager-Konfigurationsdatei und die IPMI-LAN-Konfigurationsparameter gesteuert werden können.

Momentan werden die folgenden Konfigurationsparameter unterstützt:

**TABELLE 2-2** Shelf Manager-Konfigurationsparameter

| Name | Typ | Standard-einstellung | Beschreibung |
|---|---|---|---|
| 2_X_SYSTEM | Boolescher Wert | Keine | Wenn dieser Parameter angegeben ist, bezeichnet er das aktuelle System ausdrücklich als AdvancedTCA (wenn FALSE eingestellt ist). Wenn der Parameter nicht angegeben ist (TRUE), wird die Auswahl des Systemtyps automatisch vorgenommen. Es ist nicht ratsam, diesen Parameter anzugeben, es sei denn, dadurch soll ein falscher Hardwareerkennungsalgorithmus für den Systemtyp überschrieben werden. |
| ALARM_CUTOFF_TIMEOUT | Zahl | 600 Sekunden (10 Minuten) | Das Zeitlimit für die Alarmauslösung (Zeitraum, nach dem die Alarmauslösung deaktiviert wird) in Sekunden. |
| ALLOW_CLEARING_CRITICAL_ALARM | Boolescher Wert | FALSE | Wenn Sie diesen Parameter auf TRUE setzen, können Sie einen kritischen Alarmzustand mithilfe des CLI-Befehls clia alarm clear löschen. |
| ALTERNATE_CONTROLLER | Boolescher Wert | TRUE | Verwenden des Ausweichcontrollers im Shelf Manager mit address = *ShMM-Hardwareadresse*. |
| AUTO_SEND_MESSAGE | Boolescher Wert | TRUE | Automatisches Konvertieren einer RMCP-Anforderung, die an eine andere Adresse als an eine Shelf Manager-IPMB-Adresse gesendet wurde, in eine Send Message-Anforderung, die an diese Adresse gerichtet ist. |
| CARRIER | Zeichen-folge(16) | PPS | Der Name der spezifischen Trägerkarte, auf der das ShMM installiert ist. |
| CARRIER_OPTIONS | Zeichen-folge(256) | " " | Die trägerspezifischen Optionen, die für jeden unterstützten Träger gesondert definiert sind. |
| CONSOLE_LOGGING_ENABLED | Boolescher Wert | FALSE | Ausgabeprotokollmeldungen an die Konsole, über die der Shelf Manager gestartet wurde. |
| COOLING_IGNORE_LOCAL_CONTROL | Boolescher Wert | FALSE | Kein Verwenden von lokalen Steuerfunktionen für Lüftereinrichtungen. Der Shelf Manager verwaltet die Lüftereinstellung ausdrücklich selbst. |

**TABELLE 2-2** Shelf Manager-Konfigurationsparameter *(Fortsetzung)*

| Name | Typ | Standard-einstellung | Beschreibung |
|---|---|---|---|
| COOLING_POLL_TIMEOUT | Zahl | 30 Sekunden | Der maximale Zeitraum (in Sekunden) zwischen aufeinanderfolgenden Aufrufen des Threads für die Lüftungsüberwachung und -verwaltung. |
| CTCA_FRU_RESET_TIMEOUT | Zahl | N/V | N/V |
| CTCA_HEALTHY_TIMEOUT | Zahl | N/V | N/V |
| CTCA_INITIAL_FAN_LEVEL | Zahl | N/V | N/V |
| DEFAULT_GATEWAY_IP_ADDRESS | IP-Adresse | Keine | Die standardmäßige IP-Adresse, die für das Gateway für die shelfexterne (RMCP-basierte) Kommunikation verwendet wird, wenn der entsprechende Parameter in den IPMI-LAN-Konfigurationsparametern für Kanal 1 auf `0.0.0.0` gesetzt ist. Wenn in den LAN-Konfigurationsparametern eine nicht auf Null gesetzte IP-Adresse angegeben wird, wird der in der Shelf Manager-Konfigurationsdatei bereitgestellte Wert ignoriert. |
| DEFAULT_RMCP_IP_ADDRESS | IP-Adresse | Keine | Die standardmäßige IP-Adresse für die shelfexterne (RMCP-basierte) Kommunikation. Sie wird für die redundanten Instanzen des Shelf Managers jeweils entsprechend umgeschaltet. Diese IP-Adresse wird nur verwendet, wenn der entsprechende Parameter in den IPMI-LAN-Konfigurationsparametern für Kanal 1 und im Shelf Manager-IP-Verbindungsdatensatz in den Shelf-FRU-Informationen auf `0.0.0.0` gesetzt ist. Wenn in den LAN-Konfigurationsparametern und/oder in den Shelf-FRU-Informationen eine nicht auf Null gesetzte IP-Adresse angegeben wird, wird der in der Shelf Manager-Konfigurationsdatei bereitgestellte Wert ignoriert. |
| DEVICE_POLL_TIMEOUT | Zahl | 10 Sekunden | Der Zeitraum (in Sekunden) zwischen aufeinanderfolgenden Abfragen der IPMB-Geräte durch den Shelf Manager mithilfe des Befehls `Get Device ID`. |

**TABELLE 2-2** Shelf Manager-Konfigurationsparameter *(Fortsetzung)*

| Name | Typ | Standard-einstellung | Beschreibung |
|---|---|---|---|
| EXIT_IF_NO_SHELF_FRU | Boolescher Wert | FALSE | Wenn TRUE eingestellt ist, wird der Shelf Manager beendet (und das ShMM wird ggf. zurückgesetzt), falls keine Shelf-FRU gefunden werden kann. |
| INITIAL_FAN_LEVEL | Zahl | 5 | Die ursprüngliche Lüftereinstellung, die der Shelf Manager auf Lüfterelemente anwendet. In der Regel liegen Lüftereinstellungen im Bereich 0 bis 15, wobei 0 die niedrigste und 15 die höchste mögliche Lüftergeschwindigkeit ist. |
| IPMB_ADDRESS | Zahl | 0 | Die IPMB-Adresse des Shelf Managers, die die Hardwareadresse außer Kraft setzt. Der Wert 0 bewirkt, dass der Shelf Manager die Hardwareadresse aus der Hardware ausliest und die IPMB-Adresse auf die Hardwareadresse * 2 setzt. |
| IPMB_RETRIES | Zahl | 3 | Die Anzahl an Versuchen, eine IPMB-Anforderung erneut zu senden, bevor aufgegeben wird, wenn auf die Anforderung keine Reaktion erfolgt. |
| IPMB_RETRY_TIMEOUT | Zahl | 4 Sekunden | Der Zeitraum, in dem der Shelf Manager nach dem Senden einer IPMB-Anforderung auf eine Antwort wartet, bevor das Senden der Anforderung erneut ausgeführt wird. |
| LOCAL_SHELF_FRU | Boolescher Wert | TRUE | Erstellt im Shelf Manager eine lokale FRU 1, die die Shelf-FRU-Informationen offenlegt (aus der Datei /var/nvdata/shelf_fru_info). |
| M7_TIMEOUT | Zahl | -1 (Sekunde) | Der maximale Zeitraum (in Sekunden), den eine FRU im M7-Status verbleibt. Wenn die Zeit abgelaufen ist, geht die FRU automatisch in den M0-Status über. -1 (Standardeinstellung) steht für *endlos*. Wenn Sie diesen Parameter auf 0 setzen, wird verhindert, dass FRUs den M7-Status annehmen können. |
| MAX_ALERT_POLICIES | Zahl | 64 | Die maximale Anzahl an verfügbaren PEF-Warnverfahren. |
| MAX_ALERT_STRINGS | Zahl | 64 | Die maximale Anzahl an verfügbaren PEF-Warnzeichenfolgen. |

**TABELLE 2-2** Shelf Manager-Konfigurationsparameter *(Fortsetzung)*

| Name | Typ | Standard-einstellung | Beschreibung |
|---|---|---|---|
| MAX_DEFERRED_ALERTS | Zahl | 32 | Die maximale Anzahl an ausstehenden PEF-Warnmeldungen. |
| MAX_EVENT_FILTERS | Zahl | 64 | Die maximale Anzahl an verfügbaren PEF-Ereignisfiltern. |
| MAX_EVENT_SUBSCRIBERS | Zahl | 64 | Die maximale Anzahl an Einheiten, die gleichzeitig über ein Abonnement für den Empfang von Ereignisbenachrichtigungen vom Shelf Manager verfügen können. |
| MAX_EVENT_SUBSCRIBER_IDLE_TIME | Zahl | 60 Sekunden | Der maximale Zeitraum in Sekunden für einen Ereignisabonnenten zwischen dem Zeitpunkt, zu dem ein Ereignis empfangen wird, und dem Zeitpunkt, zu dem der Abonnent dieses Ereignis aus dem Shelf Manager abruft. Wenn dieser Zeitraum überschritten wird, wird der Abonnent als nicht mehr vorhanden angesehen, und seine Registrierung wird automatisch aufgehoben. |
| MAX_PENDING_EVENT_NOTIFICATIONS | Zahl | 1024 | Die maximale Anzahl an ausstehenden Ereignisbenachrichtigungen für jeden aktiven Abonnenten. |
| MAX_PENDING_IPMB_REQUESTS | Zahl | 64 | Die maximale Anzahl an ausstehenden IPMB-Anforderungen, die auf eine Antwort warten. |
| MAX_SEL_ENTRIES | Zahl | 1024 | Die maximale Anzahl an Einträgen im Systemereignisprotokoll (System Event Log, SEL). |
| MAX_SESSIONS | Zahl | 32 | Die maximale Anzahl an gleichzeitigen IPMI-Sitzungen. |
| MAX_USERS | Zahl | 32 | Die maximale Anzahl an IPMI-Benutzern. |
| MIN_FAN_LEVEL | Zahl | 1 | Die minimale Lüftereinstellung. Der Code für die Verwaltung der Lüftung kann die Lüftereinstellung für keinen Lüfter niedriger einstellen, wenn die Lüftereinstellung automatisch erfolgt. |
| MIN_SHELF_FRUS | Zahl | 2 | Die minimale Anzahl an Shelf-FRUs im Shelf, die der Shelf Manager erkennen muss, um erfolgreich starten zu können. |

**TABELLE 2-2** Shelf Manager-Konfigurationsparameter *(Fortsetzung)*

| Name | Typ | Standard-einstellung | Beschreibung |
|---|---|---|---|
| PHYSICAL_SENSORS | Boolescher Wert | TRUE | Erstellen von IPMI-Sensoren basierend auf physikalischen Sensoren, die von ADM1026- und LM75-Chips gehostet werden. |
| POWER_UNLISTED_FRUS | Boolescher Wert | TRUE | Zulassen, dass die nicht in der Energieverwaltungstabelle der Shelf-FRU-Informationen aufgeführten FRUs aktiviert und hochgefahren werden. |
| PROPAGATE_RMCP_ADDRESS | Boolescher Wert | FALSE | Wenn der Parameter auf TRUE eingestellt ist, propagiert der aktive Shelf Manager die RMCP-IP-Adresse auf den Sicherungs-Shelf Manager. Dieser konfiguriert die Netzwerkschnittstelle, die unter Verwendung dieser IP-Adresse von der Variablen RMCP_NET_ADAPTER angegeben wird, wobei das am wenigsten signifikante Bit jedoch invertiert ist. |
| REDUNDANCY_ENABLED | Boolescher Wert | TRUE | Ausführen des Shelf Managers im redundanten Modus. |
| REDUNDANCY_NET_ADAPTER | Zeichenfolge(16) | usb0 | Der Name des Netzwerkadapters, der für die Kommunikation zwischen redundanten Instanzen des Shelf Managers verwendet wird. |
| REDUNDANCY_NET_ADAPTER2 | Zeichenfolge(16) | usb1 | Der Name des zweiten Netzwerkadapters, der für die Kommunikation zwischen redundanten Instanzen des Shelf Managers verwendet wird (wenn hierfür die Dual USB-Netzwerkschnittstelle verwendet wird). |
| REDUNDANCY_NETMASK | Zahl | 0 | Die Netzmaske, die der Redundanz-IP-Adresse zugewiesen wird. Standardmäßig (bei der Einstellung 0) wird die Netzmaske automatisch über die Klasse der IP-Adresse bestimmt. |
| REDUNDANCY_PORT | Zahl | 1040 | Der TCP-Port, der für Interaktionen zwischen redundanten Instanzen des Shelf Managers verwendet wird. |

**TABELLE 2-2** Shelf Manager-Konfigurationsparameter *(Fortsetzung)*

| Name | Typ | Standard-einstellung | Beschreibung |
|---|---|---|---|
| `REDUNDANT_IP_ADDRESS` | IP-Adresse | Keine | Die IP-Adresse, die für redundante Kommunikationsabläufe verwendet wird. Diese Adresse gibt ein IP-Adressenpaar an, bei dem die Adressen sich nur hinsichtlich des am wenigsten signifikanten Bits unterscheiden. Sie werden gemäß ihren Hardwareadressen redundanten Shelf Managern zugewiesen. |
| `RESERVATION_RETRIES` | Zahl | `10` | Die maximale Anzahl an erneuten Versuchen, die der Shelf Manager bei der Ausführung des Befehls `Reserve Device SDR` unternimmt. |
| `RMCP_NET_ADAPTER` | Zeichenfolge(16) | `eth0` | Der Name des Netzwerkadapters, der für die RMCP-basierte Kommunikation verwendet wird. |
| `RMCP_NET_ADAPTER2` | Zeichenfolge(16) | `Keine` | Der Name des anderen Netzwerkadapters, der für die RMCP-basierte Kommunikation verwendet wird, wenn die Hardware Überkreuz-Links unterstützt. |
| `SDR_READ_RETRIES` | Zahl | `3` | Die maximale Anzahl an erneuten Versuchen, die der Shelf Manager bei der Ausführung des Befehls `Read Device SDR` unternimmt. |
| `SEL_HIGH_WATERMARK` | Zahl | `0` | Das *High Watermark* für den Algorithmus, der das automatische Löschen des SEL steuert. Wenn der tatsächliche Prozentsatz der freien Einträge im SEL unter diesen Wert fällt oder wenn das SEL überläuft, startet der Shelf Manager einen Thread, der alte Datensätze absteigend nach ihrem Alter aus dem SEL löscht. |
| `SEL_LOW_WATERMARK` | Zahl | `0` | Das *Low Watermark* für den Algorithmus, der das automatische Löschen des SEL steuert. Wenn der Thread gestartet wird, der alte Datensätze aus dem SEL löscht, wird das Löschen der Datensätze so lange fortgesetzt, bis der Prozentsatz an belegten Einträgen im SEL unter diesen Wert fällt. |

**TABELLE 2-2** Shelf Manager-Konfigurationsparameter *(Fortsetzung)*

| Name | Typ | Standard-einstellung | Beschreibung |
|---|---|---|---|
| SHELF_FRU_IN_EEPROM | Boolescher Wert | TRUE | Wenn für diesen Parameter TRUE eingestellt ist, werden die Shelf-FRU-Informationen auf trägerspezifische Weise aus den EEPROMs auf der Rückwandplatine (Backplane) abgerufen. Wenn FALSE eingestellt ist, werden die Shelf-FRU-Informationen aus einer Datei im Flash-Dateisystem abgerufen. |
| SHELF_FRU_TIMEOUT | Zahl | 5 Sekunden | Das Zeitintervall während der Initialisierung, also wie lange der Shelf Manager darauf wartet, dass die Shelf-FRU-Informationsgeräte erkannt werden. |
| SHORT_SEND_MSG_RESPONSE | Boolescher Wert | TRUE | Bestimmt den Typ der Send Message-Antwort, die vom Shelf Manager bereitgestellt wird: Ist für PICMG 3.0 ECR (bei TRUE) oder für kompatible Einheiten vorheriger Versionen des Shelf Managers (bei FALSE) erforderlich. |
| SWITCHOVER_TIMEOUT_ON_BROKEN_LINK | Zahl | 10 Sekunden | Dieser Parameter bewirkt, wann bzw. ob der Shelf Manager einen Switchover (Umschaltung) initiiert, wenn der physikalische Netzwerklink zwischen dem Shelf Manager und dem System Manager (der RMCP-Link) unterbrochen ist. Wenn der Link mindestens für den Zeitraum unterbrochen bleibt, der in diesem Parameter in Sekunden angegeben ist, wird ein Switchover (Umschaltung) ausgeführt. Wenn der Link während dieses Timeout-Zeitraums wiederhergestellt wird, wird kein Switchover ausgeführt. Wenn der Wert dieses Parameters -1 lautet, werden für unterbrochene RMCP-Links keine automatischen Switchovers ausgeführt. |
| SYSLOG_LOGGING_ENABLED | Boolescher Wert | TRUE | Ausgabeprotokollmeldungen für das Systemprotokoll. |
| TASKLET_RETRIES | Zahl | 3 | Die Anzahl der Versuche, die ein Shelf Manager-Tasklet (Aktivierung, Deaktivierung, Informationsabruf) unternimmt, bevor der Versuch aufgegeben wird. |

**TABELLE 2-2** Shelf Manager-Konfigurationsparameter *(Fortsetzung)*

| Name | Typ | Standard-einstellung | Beschreibung |
|---|---|---|---|
| VERBOSITY | Zahl | 7 | Die Detailebene des Shelf Managers. |
| VERIFY_SHELF_FRU_CHECKSUM | Boolescher Wert | TRUE | Aktiviert die Überprüfung von Prüfsummen in den Datensätzen der Shelf-FRU-Informationen. Wenn der Parameter auf FALSE eingestellt ist, ignoriert der Shelf Manager Prüfsummen. |
| WATCHDOG_ENABLED | Boolescher Wert | TRUE | Verwenden des Watchdog-Timers für Hardware, der per CPLD unterstützt wird. |

Standardmäßig werden die Variablen der Konfigurationsdatei automatisch verwendet, wenn das ShMM zum ersten Mal gestartet wird. Die Standardkonfigurationsdatei importiert die folgenden Umgebungsvariablen, die von U-Boot festgelegt werden:

| | |
|---|---|
| $CARRIER_OPTIONS | Standardmäßige trägerspezifische Optionen |
| $IPADDR | Standardmäßige RMCP-IP-Adresse |
| $IPDEVICE | Standardmäßiger RMCP-Netzwerkadapter |
| $IP1ADDR | Standardmäßige redundante IP-Adresse |
| $IP1DEVICE | Standardmäßiger redundanter Netzwerkadapter |
| $GATEWAY | Standardgateway für die RMCP-Kommunikation |

Sie können den Shelf Manager bei Bedarf auf die werkseitigen Standardparameterwerte zurücksetzen. Das CODE EXAMPLE 2-1 zeigt eine Standardkonfigurationsdatei:

**CODE EXAMPLE 2-1** Standarddatei shelfman.conf

```
# /etc/shelfman.conf
#
# This is the PPS Shelf Manager configuration file.
# Copyright (c) 2005 Pigeon Point Systems.
# All rights reserved.
#

# CARRIER: This parameter is the name of the carrier-specific module to use.
#    Default is PPS.
CARRIER = $CARRIER

# CARRIER_OPTIONS: This parameter specifies the carrier-specific options.
#    Default is an empty string.
```

**CODE EXAMPLE 2-1** Standarddatei `shelfman.conf` *(Fortsetzung)*

```
CARRIER_OPTIONS = $CARRIER_OPTIONS

# ALTERNATE_CONTROLLER: This parameter of boolean type specifies whether to
#    use the alternate controller on the Shelf Manager with the address
#    equal to the ShM hardware address. Default is TRUE.
#
ALTERNATE_CONTROLLER = TRUE

# ALLOW_CLEARING_CRITICAL_ALARM: This parameter of boolean type enables the
#    ability to clear the critical alarm condition without the alarm cutoff
#    button. Default is FALSE.
#
ALLOW_CLEARING_CRITICAL_ALARM = FALSE

# ALARM_CUTOFF_TIMEOUT: This parameter specifies the time interval in
# seconds for the Shelf Manager to hold the Alarm Cutoff state. Default
# interval is 600 seconds.
#
ALARM_CUTOFF_TIMEOUT = 600

# COOLING_IGNORE_LOCAL_CONTROL: This parameter of boolean type specifies
#    whether the Shelf Manager should use local control capabilities on fan
#    devices i.e. whether the Shelf Manager should explicitly manage fan
#    levels or not. Default is FALSE.
#
COOLING_IGNORE_LOCAL_CONTROL = FALSE

# COOLING_POLL_TIMEOUT: This parameter specifies the maximum time (in
# interval is between subsequent invocations of the cooling monitoring and
# management facility. Default is 30 seconds.
#
COOLING_POLL_TIMEOUT = 30

# DEVICE_POLL_TIMEOUT: This parameter specifies the time (in seconds)
# between subsequent polls of the IPMB-0 devices by the Shelf Manager via
# sending the "Get Device ID" command to them. Default is 10 seconds.
#
DEVICE_POLL_TIMEOUT = 10

# IPMB_ADDRESS: This parameter defines the IPMB address of the Shelf
# Manager's slot. This parameter overrides the hardware address. The default
# value of0 forces the Shelf Manager to use the hardware address and set its
# IPMB address to hardware address * 2.
#
# IPMB_ADDRESS = 0
```

**CODE EXAMPLE 2-1** Standarddatei `shelfman.conf` *(Fortsetzung)*

```
# IPMB_RETRIES: This parameter is the number of attempts to re-send an IPMB
#    request before finally giving up, if no response is received to this
#    request. Default is 3.
#
IPMB_RETRIES = 3

# IPMB_RETRY_TIMEOUT: This parameter is the amount of time (in seconds) the
#    Shelf Manager waits for a response after sending an IPMB request, before
#    retrying it. Default is 4 seconds.
#
IPMB_RETRY_TIMEOUT = 4

# M7_TIMEOUT: This parameter specifies the maximum time interval (in
#   seconds for a FRU to stay in M7 state. After the expiration of this time
# the FRU automatically transitions into the M0 state. Default is -1 which
# means "forever". Setting this parameter to 0 completely prevents FRUs from
# going into the M7 state.
#
M7_TIMEOUT = -1

# MAX_ALERT_POLICIES: This parameter specifies the number of available
entries
#    in the PEF Alert Policy table. Default is 64.
#
MAX_ALERT_POLICIES = 64

# MAX_ALERT_STRINGS: This parameter specifies the number of available
entries
#    in the PEF Alert String table. Default is 64.
#
MAX_ALERT_STRINGS = 64

# MAX_DEFERRED_ALERTS: This parameter sets the maximum number of outstanding
#    PEF alerts. Default is 32.
#
MAX_DEFERRED_ALERTS = 32

# MAX_EVENT_FILTERS: This parameter specifies the number of available
entries
#    in the PEF Event Filter table.
#
MAX_EVENT_FILTERS = 64

# MAX_OEM_FILTERS: This parameter specifies the number of available entries
```

**CODE EXAMPLE 2-1** Standarddatei `shelfman.conf` *(Fortsetzung)*

```
#    in the PEF OEM Event Filter table. Default is 16.
#
MAX_OEM_FILTERS = 16

# MAX_PENDING_IPMB_REQUESTS: The parameter sets the maximum number of
#    pending IPMB requests awaiting response. Default is 192.
#
MAX_PENDING_IPMB_REQUESTS = 192

# MAX_SEL_ENTRIES: The parameter defines the SEL capacity in records.
# Default is 1024.
#
MAX_SEL_ENTRIES = 1024

# SEL_HIGH_WATERMARK: This parameter is the "high watermark" for the
algorithm
# algorithm that controls automatic SEL purging. The purging process will
# start when the actual percentage of free entries in SEL falls below this
# value or the SEL is full. During the purge the oldest SEL records are
# removed according their timestamp. Default is 10 percent i.e. start
# purging when SEL is full.
#
SEL_HIGH_WATERMARK = 10

# SEL_LOW_WATERMARK: This parameter is the "low watermark" for the algorithm
#    that controls automatic SEL purging. When the SEL purging thread starts
#    it removes records one by one until the percentage of remaining occupied
#    entries in the SEL falls below this value. Default is 50 percent.
#
SEL_LOW_WATERMARK = 50

# MAX_SESSIONS: This parameter specifies the maximum number of simultaneous
#    IPMI sessions. Default 32.
#
MAX_SESSIONS = 32

# MAX_USERS: This parameter specifies the maximum number of IPMI users.
#    Default is 32.
#
MAX_USERS = 32

# INITIAL_FAN_LEVEL: This parameter specifies the initial fan level that the
# Shelf Manager applies to fan trays. Usually fan levels values are in
# 0..15 range where 0 is the slowest, and 15 is the fastest possible fan
# speed. This parameter has an alias CTCA_INITIAL_FAN_LEVEL for CompactPCI
```

**CODE EXAMPLE 2-1** Standarddatei `shelfman.conf` *(Fortsetzung)*

```
# systems. Default is 5.
#
INITIAL_FAN_LEVEL = 5

# MIN_FAN_LEVEL: This parameter specifies the minimal fan level that can be
# set by the Cooling Management. Default is 0.
#
MIN_FAN_LEVEL = 1

# PHYSICAL_SENSORS: This parameter of boolean type specifies whether the
# Shelf Manager should create IPMI sensors based on physical sensors hosted
# by ADM1026 and LM75. Default is TRUE.
#
PHYSICAL_SENSORS = TRUE

# POWER_UNLISTED_FRUS: This parameter of boolean type specifies whether the
# Shelf Manager should power up and activate FRU devices that are not listed
# in the Power Management table of the Shelf FRU Information. Default is
# TRUE.
#
POWER_UNLISTED_FRUS = TRUE

# AUTO_SEND_MESSAGE: This parameter of boolean type specifies whether to
# auto-convert RMCP requests targeting a non-ShM IPMB address into "Send
# Message" requests directed to that address. Default is TRUE.
#
AUTO_SEND_MESSAGE = TRUE

# SHORT_SEND_MSG_RESPONSE: This parameter of boolean type determines the
# type of response on the Send Message command provided by the Shelf
# Manager:required by the PICMG 3.0 R1.0 ECN-001 if TRUE or compatible with
# previous versions of the Shelf Manager if FALSE. Default is TRUE.
#
SHORT_SEND_MSG_RESPONSE = TRUE

# SDR_READ_RETRIES: This parameter sets the number of times the Shelf
# Manager retries the "Read Device SDR" command. Default is 3.
#
SDR_READ_RETRIES = 3

# RESERVATION_RETRIES: This parameter specifies the number of times the
# Shelf Manager retries the "Reserve Device SDR" command. Default is 10.
#
RESERVATION_RETRIES = 10
```

**CODE EXAMPLE 2-1** Standarddatei `shelfman.conf` *(Fortsetzung)*

```
# TASKLET_RETRIES: This parameter specifies the number of times each Shelf
# Manager tasklet (activation, deactivation, getting information) is
# retried before finally giving up. The default is 3.
#
TASKLET_RETRIES = 3

# SHELF_FRU_IN_EEPROM: This parameter of boolean type tells the Shelf
# Manager if it should use SEEPROMs as the Shelf FRU Info storage. If set
# to FALSE the "/var/nvdata/shelf_fru_info” file contents are used. Default
# is TRUE.
#
SHELF_FRU_IN_EEPROM = TRUE

# LOCAL_SHELF_FRU: This parameter of boolean type specifies whether the
# Shelf Manager should create a local FRU#1 that will expose the Shelf FRU
# Info (obtained from the "/var/nvdata/shelf_fru_info" file). If the Shelf
# FRU Info is acquired from EEPROM as a result of the SHELF_FRU_IN_EEPROM
# set to TRUE then this parameter ignored. Default is TRUE.
#
LOCAL_SHELF_FRU = TRUE

# SHELF_FRU_TIMEOUT: This parameter specifies the time interval (in seconds)
# during which the Shelf Manager detects and reads the Shelf FRU Information
# source devices at initial startup. Default is 15 seconds.
#
SHELF_FRU_TIMEOUT = 15

# MIN_SHELF_FRUS: This parameter specifies the minimum number of valid and
# equal Shelf FRU Information instances that must be found to determine the
# true Shelf FRU Information. Default is 2.
#
MIN_SHELF_FRUS = 2

# EXIT_IF_NO_SHELF_FRU: This parameter of boolean type tells the Shelf
# Manage if it should exit if no valid Shelf FRU Information data is found.
# Default is FALSE.
#
EXIT_IF_NO_SHELF_FRU = FALSE

# VERIFY_SHELF_FRU_CHECKSUM: This parameter boolean type specifies whether
# the Shelf FRU Information record checksums should be validated. The
# default is TRUE.
#
VERIFY_SHELF_FRU_CHECKSUM = TRUE
```

**CODE EXAMPLE 2-1** Standarddatei `shelfman.conf` *(Fortsetzung)*

```
# WATCHDOG_ENABLED: This parameter of boolean type tells the Shelf Manager
# whether it should use the hardware watchdog timer supported by the CPLD or
# not. The default is TRUE.
#
WATCHDOG_ENABLED = TRUE

# REDUNDANCY_ENABLED: This parameter of boolean type tells Shelf Manager if
#    it should run in redundant mode or not. Default is TRUE.
#
REDUNDANCY_ENABLED = TRUE

# REDUNDANCY_PORT: The parameter specifies the TCP port number used for
# inter-host communications by redundant instances of the Shelf Manager.
# Default is 1040.
#
REDUNDANCY_PORT = 1040

# REDUNDANCY_NET_ADAPTER: This parameter specifies the name of network
# adapter used for communication between redundant ShMMs. Default is eth0 if
# it does not conflict with RMCP_NET_ADAPTER.
#
REDUNDANCY_NET_ADAPTER = $IP1DEVICE

# REDUNDANCY_NET_ADAPTER2: This parameter specifies the name of the second
# network adapter used for communication between redundant ShMMs (if USB
# interface is used for redundancy). By default, this parameter is not
# defined.
#REDUNDANCY_NET_ADAPTER2 = "usb1"

# REDUNDANT_IP_ADDRESS: This parameter specifies the IP address for network
# adapter used for redundant communications. This address actually provides
# a pair of IP addresses that differ in the least significant bit. They are
# assigned to redundant ShMs according to their hardware addresses, so they
# are equal on both ShMs. This parameter has no default value and must
# always be set.
#
REDUNDANT_IP_ADDRESS = $IP1ADDR

# REDUNDANCY_NETMASK: This parameter sets the network mask for the network
# adapter used for redundancy communications. Default is 255.255.255.0
#
# REDUNDANCY_NETMASK = 255.255.255.0

# RMCP_NET_ADAPTER: This parameter specifies the name of network adapter
# used for RMCP-based communications. Default is eth0:1 if it does not
```

**CODE EXAMPLE 2-1** Standarddatei `shelfman.conf` *(Fortsetzung)*

```
# conflict with REDUNDANCY_NET_ADAPTER.
#
RMCP_NET_ADAPTER = $IPDEVICE

# RMCP_NET_ADAPTER2:  This parameter specifies the alternate name of network
# adapter used for RMCP-based communications, if cross-connect links are
# supported by hardware.  Undefined by default.
#
#RMCP_NET_ADAPTER2 = "eth1"

# DEFAULT_RMCP_IP_ADDRESS: This parameter specifies the default IP address
# for network adapter used for RMCP communications. It is switched over
# between redundant instances of the Shelf Manager. This address is only
# used if no IP address is set in the LAN Configuration Parameters for
# channel # 1. Default is the REDUNDANT_IP_ADDRESS parameter value.
#
DEFAULT_RMCP_IP_ADDRESS = $RMCPADDR

# PROPAGATE_RMCP_ADDRESS: This parameter specifies whether the RMCP IP
# address should be propagated to the backup Shelf Manager. If set, the
# backup Shelf Manager configures its network interface specified by
# RMCP_NET_ADAPTER using given IP address with the least significant bit
# inverted. Default is FALSE.
#
PROPAGATE_RMCP_ADDRESS = FALSE

# DEFAULT_RMCP_NETMASK: This parameter specifies the network mask for
# network adapter used for RMCP communications. Default is 255.255.255.0
#
# DEFAULT_RMCP_NETMASK = 255.255.255.0

# DEFAULT_GATEWAY_IP_ADDRESS: This parameter specifies the default gateway
# IP address used for RMCP-based communications. It should be equal for the
# redundant instances of the Shelf Manager. This address is only used if no
# gateway address is set in the LAN Configuration Parameters for channel 1.
# Default is no gateway.
#
DEFAULT_GATEWAY_IP_ADDRESS = $GATEWAY

# SWITCHOVER_TIMEOUT_ON_BROKEN_LINK: This parameter sets the number of
# seconds to wait before switchover if the RMCP link is down, i.e. system
# manager is inaccessible from the shelf manager. A zero value of this
# parameter leads to an immediate switchover on RMCP link fault detection.
# With a -1 value, no automatic switchovers on RMCP link faults will occur.
# The default value is 10 second.
```

**CODE EXAMPLE 2-1** Standarddatei `shelfman.conf` *(Fortsetzung)*

```
#
SWITCHOVER_TIMEOUT_ON_BROKEN_LINK = 10

# CONSOLE_LOGGING_ENABLED: This parameter of boolean type enables or
# disables log messages output to the console from which the Shelf Manager
# was started. Default is FALSE.
#
CONSOLE_LOGGING_ENABLED = FALSE

# SYSLOG_LOGGING_ENABLED: This parameter of boolean type enables or disables
# logging messages to the syslog facility. Default is TRUE.
#
SYSLOG_LOGGING_ENABLED = TRUE

# VERBOSITY: This parameter sets the Shelf Manager verbosity level. This
# value is actually a bitmask with each bit enabling a corresponding class
# of output messages. The current bit layout has 8 classes:
#       Errors:                 0x01
#       Warnings:               0x02
#       Information:            0x04
#       Verbose Info:           0x08
#       Debug Trace Messages: 0x10 (not recommended)
#       Verbose Debug Trace:  0x20 (not recommended)
#       Demo Messages:          0x40 (not recommended)
#       Locks Information:    0x80 (not recommended)
# The default verbosity level is 7 i.e. errors, warnings and information.
#
VERBOSITY = 7


### PICMG 2.x specific settings

# 2_X_SYSTEM: If configured, this parameter explicitly specifies the current
# system as CompactPCI (if TRUE) or AdvancedTCA (if FALSE). If not specified
# the choice of the system type is made automatically. It is not recommended
# to specify this parameter, unless it is necessary to override an incorrect
# hardware detection algorithm for the system type. Default is FALSE.
#
# 2_X_SYSTEM = FALSE

# CTCA_FRU_RESET_TIMEOUT: This parameter specifies the time interval in
# msecs which is used to holds the BD_SEL# line low in order to reset a
# CompactPCI board. Default is 500 milliseconds.
#
# CTCA_FRU_RESET_TIMEOUT = 500
```

**CODE EXAMPLE 2-1** Standarddatei `shelfman.conf` *(Fortsetzung)*

```
# CTCA_HEALTHY_TIMEOUT: This parameter specifies the time interval in
# seconds during which the Shelf Manager waits for the HEALTHY# signal to
# appear after powering on a CompactPCI board. If the board HEALTHY# signal
# is not detected within the specified time, the Shelf Manager will
# deactivate this board. Default is 0 which means endless waiting.
#
# CTCA_HEALTHY_TIMEOUT = 0
#
### Notification settings

# MAX_EVENT_SUBSCRIBERS: The parameter defines the maximum number of
# entities
# that can simultaneously subscribe to receive event notifications
# from the Shelf Manager.
#
MAX_EVENT_SUBSCRIBERS = 64

# MAX_PENDING_EVENT_NOTIFICATIONS: The parameter defines the maximum number
# of outstanding event notifications for each active subscriber.
#
MAX_PENDING_EVENT_NOTIFICATIONS = 1024

# MAX_EVENT_SUBSCRIBER_IDLE_TIME: This parameter defines the maximum timeout
# for an event subscriber, in seconds, between the moment when an event
# arrives and the moment when the subscriber retrieves this event from the
# Shelf Manager. If this timeout is exceed, the subscriber is considered
# dead and is automatically unregistered.
#
MAX_EVENT_SUBSCRIBER_IDLE_TIME = 60
```

# Beschreibung der Detailebene

Die Detailebene ermöglicht es, dass zusätzliche Ausgaben an die Konsole oder das Systemprotokoll (Syslog) gesendet werden. Dies hängt davon ab, wie die Konfigurationsparameter `CONSOLE_LOGGING_ENABLED` und `SYSLOG_LOGGING_ENABLED` eingestellt sind. Beim Konfigurationsparameter `VERBOSITY` handelt es sich um eine hexadezimale Bitmaske, bei der jedes Bit die Ausgabe eines bestimmten Meldungstyps steuert:

| | |
|---|---|
| 0x01 | Fehlermeldungen |
| 0x02 | Warnmeldungen |
| 0x04 | Informationsmeldungen |

| | |
|---|---|
| 0x08 | Ausführliche Informationsmeldungen |
| 0x10 | Verfolgungsmeldungen |
| 0x20 | Ausführliche Verfolgungsmeldungen |
| 0x40 | Meldungen, die für wichtige Befehle angezeigt werden, die während der Initialisierung an die IPM-Controller gesendet werden |
| 0x80 | Ausführliche Meldungen zur Einrichtung und Freigabe von internen Sperren |

Die standardmäßige Debug-Ebene ist `7`. Bei dieser Einstellung können Fehler-, Warn- und Informationsmeldungen ausgegeben werden.

# Einstellen von Datum und Uhrzeit

Wenn Sie das System zum ersten Mal starten, ist die Uhr noch nicht eingestellt und muss initialisiert werden. Die Uhr ist anfänglich auf den 1. Januar 1970 gesetzt. Sie können über die serielle Konsole auf das Datum zugreifen.

```
# date
Thu Jan  1 03:16:30 UTC 1970
```

Geben Sie das richtige Datum ein, indem Sie zum Ändern des Datums die Anwendung `date` verwenden. Das Format für den Befehl `date` lautet MMDDHHMMSSYYYY. Dabei gilt Folgendes:

| | |
|---|---|
| MM | Monat |
| DD | Tag |
| HH | Stunde (24-Stunden-Uhr) |
| MM | Minute |
| SS | Sekunde |
| YYYY | Jahr |

Beispiel:

```
# date 04291628002003
Tue Apr 29 16:28:00 UTC 2003
```

Um das Datum dauerhaft zu sichern, speichern Sie es mithilfe der Anwendung hwclock.

```
# hwclock -systohc
```

In einigen Fällen wird ggf. die folgende Fehlermeldung angezeigt:

```
mktime: cannot convert RTC time to UNIX time
```

Sie können diesen Fehler ignorieren. Die Ursache der Meldung besteht darin, dass sich das ursprüngliche Datum in einem nicht initialisierten Zustand befindet.

# Abrufen von Datum und Uhrzeit von einem Zeitserver

Wenn die Shelf-Verwaltungskarte nicht über eine akkugepufferte Echtzeituhr (Real-Time Clock, RTC) verfügt, können Sie das Systemdatum und die -uhrzeit während des Startvorgangs von einem Zeitserver abrufen und danach regelmäßig synchronisieren. Der ausgewählte Zeitserver muss RFC 868 over TCP unterstützen, da dies Voraussetzung für die Verwendung des Dienstprogramms rdate ist. Zum Aktivieren dieser Funktion müssen Sie die U-Boot-Variable time_server und wahlweise auch die zusätzliche Variable timezone definieren.

Die Variable time_server enthält die IP-Adresse des Zeitservers, auf dem der Shelf Manager nach dem Start die Systemzeit abfragen soll. Diese Variable wird als Umgebungsvariable TIMESERVER auf die Linux-Ebene propagiert. Wenn diese Variable eingerichtet ist, startet das Startskript /etc/netconfig das Skript /etc/timesync als Dämonen, der als Endlosschleife ausgeführt wird und den Zeitserver mit einem Standardintervall von 300 Sekunden abfragt. Wenn Sie dieses Intervall ändern möchten, bearbeiten Sie das Skript /etc/timesync und ändern den Wert der Variablen INTERVAL.

---

**Hinweis –** Wenn die Variable time_server angegeben ist, muss die Variable ip1device auf usb0 gesetzt sein, um eine ordnungsgemäße Synchronisierung zu ermöglichen.

---

Die Variable timezone enthält den Namen der aktuellen Zeitzone gefolgt von der Differenz zur Greenwich Mean Time (GMT). Die Differenz ist für Zeitzonen westlich von Greenwich positiv und für Zeitzonen östlich von Greenwich negativ. Diese

Variable wird als Umgebungsvariable TZ auf die Linux-Ebene propagiert. Der Standardwert dieser Variablen lautet UTC0, also Universal Coordinated Time (UTC), die der Greenwich-Zeit entspricht.

Die von Zeitservern gesendete Zeit ist GMT. Wenn die Zeitzone für den Shelf Manager nicht oder nicht richtig eingerichtet ist, wird die vom Zeitserver abgerufene Zeit nicht richtig interpretiert. Der drei Buchstaben umfassende Name der Zeitzone wird vom Shelf Manager nicht verwendet, sondern wird propagiert, um die Linux-Zeitzone festzulegen. (Wenn z. B. der Zeitzonenname XXX0 verwendet wird, bewirkt der Befehl date eine Ausgabe wie Thu Sep 9 21:24:24 XXX 2004.) Die Verwendung der Sommerzeit wird nicht unterstützt.

Das folgende Beispiel zeigt eine timezone-Definition für US Eastern Time:

```
timezone = EST5
```

Die Zahl 5 zeigt hier an, dass sich die Zeitzone fünf Stunden westlich der GMT befindet. Die Buchstaben EST können durch drei beliebige andere Buchstaben ersetzt werden. Sie werden z. B. verwendet, um die Zeitzone in Linux-Befehlsausgaben des Befehls date zu identifizieren.

# Einrichten von Benutzerkonten auf der Shelf-Verwaltungskarte

Benutzerkonten für den RMCP-Zugriff werden mithilfe des Shelf Manager-CLI eingerichtet. Sie geben die Benutzerinformationen auf der aktiven Shelf-Verwaltungskarte ein, die dann auf der Standby-Shelf-Verwaltungskarte sofort *gespiegelt* bzw. bereitgestellt werden. Die Shelf-Verwaltungskarte unterstützt 32 Konten mit Passwörtern.

## ▼ So fügen Sie ein Benutzerkonto für den RMCP-Zugriff hinzu

1. **Melden Sie sich bei der aktiven Shelf-Verwaltungskarte an.**

2. **Fügen Sie einen Benutzer hinzu:**

```
# clia user add benutzernummer benutzername kanalzugriffs-flags berechtigungsebene passwort
```

Die Variablenparameter haben hierbei folgende Bedeutung:

*benutzernummer* – gültige Benutzernummer

*benutzername* – Benutzername (bis zu 16 Zeichen)

*kanalzugriffs-flag* – jeweils das erste Byte des Befehls `SetUserInfo` (nur die Bits 4, 5 und 6 sind aussagekräftig)

- Bit 6 – IPMI-Messaging aktiviert
- Bit 5 – Linkauthentifizierung aktiviert
- Bit 4 – Auf Rückmeldung beschränkt

*berechtigungsebene* – Berechtigungsebene des Benutzers

*passwort* – Benutzerpasswort (wird ohne Meldung auf 16 Zeichen beschränkt)

Das folgende Beispiel zeigt, wie Sie den Benutzer „User 9" mit dem Namen `root`, Administrator-Berechtigungsebene und dem Passwort `PICMG guru` hinzufügen.

```
# clia user add 9 "root" 0x40 4 "PICMG guru"
IPM Sentry Shelf Manager Command Line Interpreter
User 9 added successfully
#
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
9: "root"
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
```

Weitere Informationen zu Berechtigungen und zum Befehl `clia user` finden Sie unter „`user`" auf Seite 267.

## Beschränkungen bei Benutzernamen

Das Feld für den Benutzernamen kann maximal 16 Zeichen enthalten. Außerdem muss der Benutzername mindestens einen Kleinbuchstaben enthalten, und das erste Zeichen muss ein Buchstabe sein.

Folgende Zeichen sind für das Feld *username* zulässig:

- Buchstaben
- Zahlen
- Punkt (.)
- Unterstrich (_)
- Bindestrich (-)

## Passwörter

Passwörter können maximal 16 Zeichen lang sein. Alle Zeichen danach werden abgeschnitten.

# Konfigurieren von OpenHPI im Shelf Manager

Der Shelf Manager unterstützt OpenHPI. Hierbei handelt es sich um eine Open Source-Implementierung des Hardware Platform Interface (HPI) des SA Forums. HPI stellt eine Schnittstelle zum Verwalten von Computerhardware dar, vor allem für Chassis- und Rack-Server. Der Zugriff auf das HPI erfolgt über den OpenHPI-SNMP-Subagent, der die SNMP MIB verwendet.

Es gibt zwei Konfigurationsdateien, die Systemadministratoren kennen müssen:

- `/etc/openhpi.conf` – OpenHPI-Konfigurationsdatei
- `/etc/snmpd.conf` – SNMP-Subagent-Konfigurationsdatei

## Die Datei `/etc/openhpi.conf`

Sie müssen die OpenHPI-Konfigurationsdatei `/etc/openhpi.conf` aktualisieren, um die richtige IP-Adresse für das ShMM anzugeben. Nachdem Sie die Konfigurationsdatei aktualisiert haben, setzen Sie das ShMM zurück, um die Änderungen zu implementieren.

## ▼ So ändern Sie die Datei /etc/openhpi.conf

1. **Bearbeiten Sie die Datei** `/etc/openhpi.conf`**, und ändern Sie den Wert des Parameters** `libipmdirect addr` **in die IP-Adresse des ShMM.**

   Es wird die Beispieldatei `/etc/openhpi.conf` angezeigt.

```
OPENHPI_THREADED = "YES"
OPENHPI_UID_MAP = "/var/bin/uip_map"
plugin libipmidirect
handler libipmidirect {
    entity_root = "{SYSTEM_CHASSIS, 1}"
    name = "lan"
    addr = "____________________"
    port = "623"
    auth_type = "{none"
    auth_level = "admin"
    username = "openhpi"
    password = "openhpi"
    MaxOutstanding = "1"
    ActConnectionTimeout = "5000"
    logflags = " "
    logfile = "log"
    logfile_max = "10"
    }
```

2. **Starten Sie das ShMM neu, indem Sie an der Eingabeaufforderung den Befehl** `reboot` **ausführen.**

   Beispiel:

```
# reboot
```

# Die Datei /etc/snmpd.conf

Die SNMP-Subagent-Konfigurationsdatei `/etc/snmpd.conf` definiert, wie der SNMP-Subagent arbeitet, und enthält Anweisungen für die Zugriffssteuerung und das Einrichten von Unterbrechungen. Informationen zur Zugriffssteuerung, zur SNMPv3-Konfiguration und zum Einrichten von Unterbrechungen finden Sie in den folgenden Abschnitten.

## Zugriffssteuerung

Der SNMP-Subagent unterstützt das View-Based Access Control Model (VACM), das unter RFC 2575 definiert ist. Dazu erkennt er in der Konfigurationsdatei die folgenden Schlüsselwörter:

- `com2sec`
- `group`
- `access`
- `view`

Zusätzlich werden einige einfacher anzuwendende Wrapper-Anweisungen erkannt:

- `rocommunity`
- `rwcommunity`
- `rouser`
- `rwuser`

In diesem Abschnitt wird definiert, wie Sie das Programm `snmpd` konfigurieren, damit verschiedene Zugriffstypen und -ebenen akzeptiert werden.

`rouser` *benutzer* `[noauth|auth|priv] [`*OID*`]`
`rwuser` *benutzer* `[noauth|auth|priv] [`*OID*`]`

Erstellt in den VACM-Zugriffskonfigurationstabellen einen SNMPv3 USM-Benutzer. Es ist effizienter (und leistungsfähiger), die kombinierten Anweisungen `group`, `access` und `view` zu verwenden, aber diese Wrapper-Anweisungen sind bedeutend einfacher anzuwenden.

Die minimale Authentifizierungs- und Vertraulichkeitsebene, die Benutzer verwenden müssen, wird durch das erste Token angegeben (Standardeinstellung `auth`). Der Parameter *OID* beschränkt den Zugriff für den jeweiligen Benutzer auf den Bereich unterhalb der angegebenen *OID*.

`rocommunity` *community* `[`*quelle*`] [`*OID*`]`
`rwcommunity` *community* `[`*quelle*`] [`*OID*`]`

Hiermit werden Communities mit und ohne Schreibschutz erstellt, die Sie verwenden können, um auf den Agenten zuzugreifen. Es handelt sich hierbei um einen schnellen Wrapper für die komplexeren und leistungsfähigeren Anweisungszeilen `com2sec`, `group`, `access` und `view`. Die Communities sind nicht so effizient, weil keine Gruppen erstellt werden und die Tabellen somit ggf. größer sind. Für komplexe Umgebungen sind diese Anweisungen nicht zu empfehlen. Sie können diese Anweisungen verwenden, wenn Ihre Umgebung relativ einfach ist oder wenn diese eine kleine Leistungsbeeinträchtigung verkraften kann.

Das Format des Tokens *source* ist unten im Abschnitt zur Anweisung `com2sec` beschrieben. Das Token *OID* beschränkt den Zugriff für die Community auf den Bereich unterhalb der angegebenen *OID*.

`com2sec` *name quelle community*

Gibt die Zuordnung von einem *quelle*/*community*-Paar zu einem Sicherheits*namen* an. `quelle` kann ein Hostname, ein Teilnetz oder das Wort `default` sein. Ein Teilnetz kann als IP/Maske oder IP/Bits angegeben werden. Es wird die erste *quelle*/*community*-Kombination ausgewählt, die mit dem eingehenden Paket übereinstimmt.

`group` *name modell sicherheit*

Definiert die Zuordnung von *sicherheitsmodell*/*sicherheitsname* zu einer `group` (Gruppe). *modell* hat entweder den Wert `v1`, `v2c` oder `usm`.

`access` *name kontext modell ebene präfix lesen schreiben benachrichtigen*

Führt die Zuordnung von den Ebenen *gruppe*/*sicherheit* und *modell*/*sicherheit* zu einer Ansicht (view) durch. *modell* hat den Wert `any`, `v1`, `v2c` oder `usm`. *ebene* hat den Wert `noauth`, `auth` oder `priv`. *präfix* gibt an, wie für *kontext* die Prüfung der Übereinstimmung mit dem Kontext der eingehenden PDU erfolgt, entweder `exact` oder `prefix`. Mithilfe der Angaben für *lesen*, *schreiben* und *benachrichtigen* wird die Ansicht festgelegt, die für den entsprechenden Zugriff verwendet wird. Beim v1- oder v2c-Zugriff hat *ebene* den Wert `noauth`, und *kontext* ist leer.

`view` *name typ teilstruktur* [*maske*]

Definiert die benannte Ansicht. *typ* hat entweder den Wert `included` oder `excluded`. Bei *maske* handelt es sich um eine Liste von sog. „Hex Octets", die durch Punkte (.) oder Doppelpunkte (:) voneinander getrennt sind. Die Maske wird standardmäßig auf `ff` gesetzt, wenn keine Angabe erfolgt. Indem Sie die *maske* verwenden, können Sie auf relativ einfache Weise den Zugriff auf eine Zeile in einer Tabelle steuern. Als Dienstanbieter können Sie z. B. jedem Kunden den Zugriff auf seine eigene Schnittstelle gewähren:

```
view cust1 included interfaces.ifTable.ifEntry.ifIndex.1 ff.a0
view cust2 included interfaces.ifTable.ifEntry.ifIndex.2 ff.a0

# interfaces.ifTable.ifEntry.ifIndex.1 == .1.3.6.1.2.1.2.2.1.1.1
# ff.a0 == 11111111.10100000
```

Diese Einträge decken den Zeilenindex ab und enthalten ihn, ermöglichen Benutzern jedoch trotzdem das Ändern des Felds einer Zeile.

Unten sind einige VACM-Beispiele aufgeführt:

```
# sec.name source community
com2sec local localhost private
com2sec mynet 10.10.10.0/24 public
com2sec public default public

# sec.model sec.name
group mygroup v1 mynet
group mygroup v2c mynet
group mygroup usm mynet
group local v1 local
group local v2c local
group local usm local
group public v1 public
group public v2c public
group public usm public

# incl/excl subtree mask
view all included .1 80
view system included system fe
view mib2 included .iso.org.dod.internet.mgmt.mib-2 fc

# context sec.model sec.level prefix read write notify
access mygroup "" any noauth exact mib2 none none
access public "" any noauth exact system none none
access local "" any noauth exact all all all
```

## SNMPv3-Konfiguration

`engineID` *zeichenfolge*

Der Subagent muss mit einer `engineID` konfiguriert werden, um auf SNMPv3-Meldungen antworten zu können. Mit dieser Zeile der Konfigurationsdatei wird die `engineID` über eine *zeichenfolge* konfiguriert. Der Standardwert der `engineID` wird mit der ersten IP-Adresse konfiguriert, die für den Hostnamen des Computers gefunden wird.

`createUser` *benutzername* `(MD5|SHA)` *authpassphrase* `[DES]`[*privpassphrase*]

`MD5` und `SHA` sind die zu verwendenden Authentifizierungstypen, aber Sie müssen das Paket mit installiertem OpenSSL erstellt haben, um SHA verwenden zu können. Das einzige Vertraulichkeitsprotokoll, das momentan unterstützt wird, ist `DES`. Wenn die *privpassphrase* nicht angegeben ist, wird angenommen, dass sie mit der *authpassphrase* identisch ist.

**Hinweis –** Die erstellten Benutzer sind nur verwendbar, wenn sie auch den oben beschriebenen VACM-Zugriffssteuerungstabellen hinzugefügt werden.

**Hinweis –** Die Mindestlänge für die Passphrase beträgt acht Zeichen.

## Einrichten von Unterbrechungen und Informationszielen

`trapcommunity` *zeichenfolge*

Definiert die Standard-Community *zeichenfolge*, die beim Senden von Unterbrechungen verwendet wird. Beachten Sie, dass Sie diesen Befehl vor den drei Befehlen (die unmittelbar folgen) verwenden müssen, die für diese Community (Zeichenfolge) verwendet werden sollen.

`trapsink` *host* [*community* [*port*]]
`trap2sink` *host* [*community* [*port*]]
`informsink` *host* [*community* [*port*]]

Definiert die Hosts, damit diese Unterbrechungen empfangen können (oder über `informsink` Informationsbenachrichtigungen). Der Dämon sendet beim Starten eine Kaltstart-Unterbrechung (Cold Start). Er sendet auch bei Authentifizierungsfehlern Unterbrechungen, falls diese Funktion aktiviert ist. Sie können mehrere `trapsink`-, `trap2sink`- und `informsink`-Zeilen für mehrere Ziele angeben. Verwenden Sie `trap2sink`, um SNMPv2-Unterbrechungen zu senden, und `informsink`, um Informationsbenachrichtigungen zu senden. Wenn für *community* keine Angabe vorhanden ist, wird die Zeichenfolge einer vorherigen `trapcommunity`-Anweisung verwendet. Wenn keine Angabe für *port* vorhanden ist, wird der bekannte SNMP-Unterbrechungs-Port (162) verwendet.

`trapsess` [*snmpcmdargs*] *host*

Hierbei handelt es sich um ein allgemeineres Token zur Unterbrechungskonfiguration, mit dem für alle SNMP-Versionen beliebige Unterbrechungsziele angegeben werden können. Dazu ist es erforderlich, dass Sie zusätzlich die Versionsnummer `v2c` oder `v3` angeben.

## ▼ So aktualisieren Sie die Datei `/etc/snmpd.conf`

1. **Bearbeiten Sie die Datei `/etc/snmpd.conf`, um Anweisungen je nach Bedarf hinzuzufügen, zu ändern oder zu löschen.**

2. **Starten Sie das ShMM neu, indem Sie an der Eingabeaufforderung den Befehl `reboot` ausführen.**

KAPITEL **3**

# Verwalten des Systems

Sie verwalten Ihr System mithilfe der Befehlszeilenoberfläche (CLI) der Shelf-Verwaltungskarte oder über Ethernet mithilfe der RMCP-Oberfläche.

Dieses Kapitel enthält folgende Abschnitte:



## IPMI LAN-Schnittstelle

Die IPMI LAN-Schnittstelle wird von der ATCA-Spezifikation gefordert und unterstützt die IPMI-Meldungsübertragung mit dem Shelf Manager über das Remote Management Control Protocol (RMCP). Ein Systemadministrator, der RMCP für die Kommunikation mit Shelves verwendet, kann mit jedem mit ATCA kompatiblen Shelf Manager kommunizieren. Diese Schnittstelle auf niedriger Ebene ermöglicht den Zugriff auf die IPMI-Funktionen eines Shelfs. Dazu zählt die Möglichkeit für den Systemadministrator, IPMI-Befehle an IPM-Controller im Shelf auszugeben, wobei der Shelf Manager als Proxy verwendet wird.

## IPMI-Befehle

Die standardmäßigen IPMI-Befehle sind in der ATCA-Spezifikation von PICMG 3.0 dokumentiert. Diese Spezifikation bietet außerdem benutzerdefinierte OEM IPMI-Befehle. Sun Microsystems hat eine einzigartige Gruppe dieser Befehle für die von Sun entwickelten ATCA-Platinen bereitgestellt. Diese Befehle sind in TABELLE 3-1 aufgelistet und werden im Anhang B beschrieben.

**TABELLE 3-1** Sun OEM IPMI-Befehle

| Befehle | Op.-Code | Syntax |
|---|---|---|
| `Get Version` | 0x80 | #GET_VERSION |
| `Set Boot Page`† | 0x81 | #SET_BOOT_PAGE |
| `Get Boot Page`† | 0x82 | #GET_BOOT_PAGE |
| `Set Front panel reset button state` | 0x83 | #SET_FP_RESET_BUTTON |
| `Get Front panel reset button state` | 0x84 | #GET_FP_RESET_BUTTON_STATE |
| `Set Ethernet Force Front bit` | 0x85 | #SET_ETH_FORCE_FRONT |
| `Get Ethernet Force Front bit` | 0x86 | #GET_ETH_FORCE_FRONT |
| `Get RTM status` | 0x88 | #GET_RTM_PRESENCE |

† Nur gültig für Sun Netra™ CP3010-Nodeboard

# Befehlszeilenoberfläche von Shelf Manager

Die Befehlszeilenoberfläche (CLI) von Shelf Manager kann verwendet werden, um mit den intelligenten Management-Controllern des Shelfs, mit Platinen und mit dem Shelf Manager selbst über Textbefehle zu kommunizieren. Die CLI ist eine IPMI-basierte Gruppe von Befehlen, auf die direkt oder über eine Management-Anwendung auf höherer Ebene oder über ein Skript zugegriffen werden kann. Administratoren können auf die CLI über eine Telnet-Verbindung oder über den seriellen Anschluss der Shelf-Verwaltungskarte zugreifen. Mithilfe der CLI können Bediener auf Informationen über den aktuellen Zustand des Shelfs einschließlich der aktuellen FRU-Bestückung, aktueller Sensorwerten, Schwellenwerteinstellungen, kürzlichen Ereignissen und des Gesamtzustands des Shelfs zugreifen.

# Starten der Befehlszeilenoberfläche

Um die CLI zu verwenden, melden Sie sich im Linux-Betriebssystem bei der Shelf-Verwaltungskarte an. Nach der Anmeldung führen Sie die EXE-Datei `clia` über die Befehlszeile mit spezifischen Parametern aus. Der erste Parameter ist das Befehlsverb. Die EXE-Datei `clia` befindet sich im virtuellen `root`-Dateisystem, das von Linux verwaltet wird und auf der Shelf-Verwaltungskarte ausgeführt wird. Die EXE-Datei `clia` stellt eine Verbindung zum Hauptprozess der Shelf Manager-Software her, leitet die Befehlsinformationen dorthin weiter und ruft die Ergebnisse ab. Der Shelf Manager muss ausgeführt werden, bevor die CLI gestartet wird.

Beispiel:

```
# clia ipmc

IPM Sentry Shelf Manager Command Line Interpreter

20: Entity: (d0, 0) Maximum FRU device ID: 20
    PICMG Version 2.0
    Hot Swap State: M4, Previous: M3, Last State Change Cause: Normal State
Change (0)
#
```

Beim Starten ohne Parameter wechselt `clia` in einen interaktiven Modus. In diesem Modus gibt das Programm wiederholt eine Eingabeaufforderung an das Terminal aus, akzeptiert Benutzereingaben als den nächsten Befehl mit Parametern, führt diesen Befehl aus und zeigt die Ergebnisse am Terminal an, bis der Benutzer den Befehl `exit` oder `quit` eingibt. Beispiel:

```
# clia

IPM Sentry Shelf Manager Command Line Interpreter

CLI> ipmc 20

20: Entity: (d0, 0) Maximum FRU device ID: 20
    PICMG Version 2.0
    Hot Swap State: M4, Previous: M3, Last State Change Cause: Normal State
Change (0)

CLI> exit
#
```

# CLI-Befehle

Die CLI implementiert die folgenden Befehle. Sie werden ausführlich in Anhang A, unter „Shelf Manager-CLI-Befehle“ auf Seite 125 mit einem Teilabschnitt für jeden Befehl in alphabetischer Reihenfolge der Befehlsnamen beschrieben.

**TABELLE 3-2** Übersicht über die CLI-Befehle für Shelf Manager

| Befehl | Parameter | Beschreibung |
|---|---|---|
| `activate` | IPMB-Adresse<br>FRU-Geräte-ID | Aktiviert die angegebene FRU. |
| `alarm` | Alarmtyp | Aktiviert oder deaktiviert einen Telco-Alarm. |
| `board` | Steckplatznummer (optional) | Zeigt Informationen über Platinen an. |
| `boardreset` | Steckplatznummer | Setzt die angegebene ATCA-Platine zurück. |
| `busres` | Unterbefehl, mit Parametern | Führt die angegebene Operation für die durch Bused E-Keying verwalteten Ressourcen durch. |
| `console` | Steckplatznummer | Öffnet eine Konsolensitzung für das Node Board im angegebenen Steckplatz. |
| `deactivate` | IPMB-Adresse<br>FRU-Geräte-ID | Deaktiviert die angegebene FRU. |
| `debuglevel` | neue Debug-Ebene (optional) | Ruft die aktuelle Debug-Ebene für den Shelf Manager ab oder legt eine neue Debug-Ebene fest. |
| `exit/quit` | | Beendet den Interpreter im interaktiven Modus. |
| `fans` | IPMB-Adresse (optional)<br>FRU-Geräte-ID (optional) | Zeigt Informationen über Lüfter an. |
| `flashupdate` | IP-Adresse des Servers<br>Pfadname zum Firmware-Abbild | Führt Download und Aktualisierung der System-Firmware auf dem Netra CP3060-Nodeboard vom angegebenen Server und Pfadnamen durch. |
| `fru` | IPMB-Adresse (optional)<br>FRU-Geräte-ID (optional)<br>FRU-Typ (optional) | Zeigt Informationen über eine FRU oder eine Gruppe von FRUs im Shelf an. FRUs werden nach Typ oder nach übergeordnetem IPM-Controller ausgewählt. |

**TABELLE 3-2** Übersicht über die CLI-Befehle für Shelf Manager *(Fortsetzung)*

| Befehl | Parameter | Beschreibung |
|---|---|---|
| `frucontrol` | IPMB-Adresse<br>FRU-Geräte-ID<br>Option | Sendet FRU Control-Befehle an spezifisches FRU. |
| `frudata` | IPMB-Adresse (optional)<br>FRU-Geräte-ID (optional)<br>Block/Byte-Offset (optional)<br>Daten (optional) | Bietet Raw-Zugriff auf FRU-Informationen für die angegebene FRU. |
| `frudatar` | IPMB-Adresse<br>FRU-Geräte-ID<br>Dateiname | Liest den FRU-Datenbereich der angegebenen FRU und speichert die Daten in der angegebenen Datei. |
| `frudataw` | IPMB-Adresse<br>FRU-Geräte-ID<br>Dateiname | Schreibt die FRU-Daten in die angegebene Datei in den FRU-Datenbereich der angegebenen FRU. |
| `fruinfo` | IPMB-Adresse<br>FRU-Geräte-ID | Bietet eine benutzerfreundliche FRU-Informationsausgabe. |
| `getfanlevel` | IPMB-Adresse (optional)<br>FRU-Geräte-ID (optional) | Zeigt die aktuelle Stufe des vom angegebenen FRU gesteuerten Lüfters an. |
| `getfruledstate` | IPMB-Adresse (optional)<br>FRU-Geräte-ID (optional)<br>LED-ID oder `ALL` (optional) | Zeigt den FRU-LED-Status an. |
| `gethysteresis` | IPMB-Adresse (optional)<br>Sensorname (optional)<br>Sensornummer (optional) | Zeigt sowohl die positive als auch die negative Hysterese des angegebenen Sensors an. |
| `getipmbstate` | IPMB-Adresse<br>IPMB-Linknummer (optional) | Zeigt den aktuellen Status von IPMB-0 bei der Zieladresse an. Wenn eine Linknummer angegeben ist und das Ziel-IPMC ein IPMB-Hub ist, werden Informationen über einen spezifischen Link angezeigt. |
| `getlanconfig` | Kanalnummer<br>Parametername oder -nummer (optional)<br>Auswahlwerkzeug festlegen (optional) | Ruft einen LAN-Konfigurationsparameter für einen spezifischen Kanal ab und zeigt diesen an. |

**TABELLE 3-2** Übersicht über die CLI-Befehle für Shelf Manager *(Fortsetzung)*

| Befehl | Parameter | Beschreibung |
|---|---|---|
| `getpefconfig` | Parametername oder -nummer (optional)<br>Auswahlwerkzeug festlegen (optional) | Ruft einen PEF-Konfigurationsparameter ab und zeigt diesen an. |
| `getsensoreventenable` | IPMB-Adresse (optional)<br>Sensorname (optional)<br>Sensornummer (optional) | Zeigt die aktuellen Sensorereignis-Maskenwerte für die unterstützten Ereignisse der angegebenen Sensoren an. |
| `getthreshold`, `threshold` | IPMB-Adresse (optional)<br>Sensorname (optional)<br>Sensornummer (optional) | Zeigt Schwellenwertinformationen über einen spezifischen Sensor an. |
| `help` | | Zeigt die Liste der unterstützten Befehle an. |
| `ipmc` | IPMB-Adresse (optional) | Zeigt Informationen über einen oder alle IPM-Controller im Shelf an. |
| `localaddress` | | Ruft die IPMB-Adresse des aktuellen Shelf Managers an. |
| `minfanlevel` | Lüfterstufe (optional) | Zeigt oder legt die minimale Lüfterstufe fest. |
| `sel` | IPMB-Adresse (optional)<br>Anzahl der Elemente (optional) | Zeigt mehrere der neusten Elemente im Systemereignisprotokoll an, das auf dem Ziel-IPM-Controller verwaltet wird. |
| `sensor` | IPMB-Adresse (optional)<br>Sensorname (optional)<br>Sensornummer (optional) | Zeigt Informationen über eine Gruppe von Sensoren an. Sensoren werden anhand der Adresse, der Nummer oder des Namens des IPM-Controllers ausgewählt. |
| `sensordata` | IPMB-Adresse (optional)<br>Sensorname (optional)<br>Sensornummer (optional) | Zeigt Wertinformationen für einen spezifischen Sensor an. |
| `sensorread` | IPMB-Adresse<br>Sensornummer | Zeigt Rohwertinformationen für einen spezifischen Sensor an (wobei den Sensor beschreibende Sensordatenaufzeichnungen ignoriert werden). |
| `session` | | Zeigt Informationen über aktive RMCP-Sitzungen an. |

**TABELLE 3-2** Übersicht über die CLI-Befehle für Shelf Manager *(Fortsetzung)*

| Befehl | Parameter | Beschreibung |
|---|---|---|
| `setextracted` | IPMB-Adresse<br>FRU-Geräte-ID | Meldet dem Shelf Manager, dass die angegebene FRU aus dem Shelf physisch herausgezogen wurde. |
| `setfanlevel` | IPMB-Adresse<br>FRU-Geräte-ID<br>Stufe | Legt eine neue Stufe des von der angegebenen FRU gesteuerten Lüfters fest. |
| `setfruledstate` | IPMB-Adresse<br>FRU-Geräte-ID<br>LED-ID oder `ALL`<br>LED-Betrieb<br>LED-Farbe (optional) | Legt den Status einer angegebenen LED oder aller LEDs für die angegebene FRU fest. |
| `sethysteresis` | IPMB-Adresse<br>Sensorname oder Sensornummer<br>Festzulegende Hysterese (`pos` oder `neg`)<br>Hysteresewert | Legt einen neuen Hysteresewert für den angegebenen Sensor fest. |
| `setipmbstate` | IPMB-Adresse<br>IPMB-Busname (`A` oder `B`)<br>IPMB-Linknummer (optional)<br>Durchzuführende Aktion | Deaktiviert/aktiviert IPMB-A oder IPMB-B (oder den spezifischen IPMB-Link) für den Ziel-IPM-Controller. |
| `setlanconfig` | Kanal<br>Parametername oder -nummer<br>Zusätzliche Parameter | Legt den Wert des LAN-Konfigurationsparameters für den angegebenen Kanal fest. |
| `setlocked` | IPMB-Adresse<br>FRU-Geräte-ID<br>Status | Legt für das Locked-Bit für die angegebene FRU den angegebenen Status fest (`0` – entsperrt, `1` – gesperrt). |
| `setpefconfig` | Parametername oder -nummer<br>Auswahlwerkzeug festlegen (optional)<br>Parameterwert | Legt einen neuen Wert eines PEF-Konfigurationsparameters fest. |
| `setpowerlevel` | IPMB-Adresse<br>FRU-Geräte-ID<br>Betriebsstatus oder `AUS`<br>`Copy` | Legt den Betriebsstatus der angegebenen FRU fest, schaltet die FRU aus und kopiert den gewünschten Status zu dem vorhandenen Status. |

**TABELLE 3-2** Übersicht über die CLI-Befehle für Shelf Manager *(Fortsetzung)*

| Befehl | Parameter | Beschreibung |
|---|---|---|
| `setsensoreventenable` | IPMB-Adresse<br>Sensorname<br>Sensornummer<br>Globale Flags<br>Maske für Bestätigungsanweisung (optional)<br>Maske für Rücknahme der Bestätigungsanweisung (optional) | Ändert die Ereignisaktivierungsmasken für einen spezifischen Sensor. |
| `setthreshold` | IPMB-Adresse<br>Sensorname<br>Sensornummer<br>Schwellenwerttyp<br>Schwellenwert | Ändert einen spezifischen Schwellenwert (oberer/unterer Wert, kritisch/nicht kritisch/nicht wiederherstellbar) für einen spezifischen Sensor. |
| `shelf` | Unterbefehl, mit Parametern | Zeigt allgemeine Informationen über das Shelf an. Mithilfe verschiedener Unterbefehle können Shelf-Attribute festgelegt und zusätzliche Informationen über bestimmte Bereiche abgerufen werden. |
| `shelfaddress` | Shelf-Adresszeichenfolge (optional) | Ruft das Shelf-Adressfeld der Adresstabelle in den Shelf-FRU-Informationen ab oder legt es fest. |
| `shmstatus` | | Zeigt den Aktiv/Backup-Status des Shelf Managers an. |
| `showhost` | Steckplatznummer | Zeigt die Versionsinformationen über die Firmware einer Netra CP3060-Platine an. |
| `showunhealthy` | | Zeigt die Komponenten des Shelfs an, die sich in einem schlechten Zustand befinden. |
| `switchover` | | Initiiert einen Wechsel zum Backup-Shelf Manager. |

**TABELLE 3-2** Übersicht über die CLI-Befehle für Shelf Manager *(Fortsetzung)*

| Befehl | Parameter | Beschreibung |
|---|---|---|
| `terminate` | | Beendet den Shelf Manager, ohne die Shelf-Verwaltungskarte neu zu starten. |
| `user` | Unterbefehl, mit Parametern | Zeigt Informationen über die RMCP-Benutzerkonten im Shelf Manager an und ermöglicht das problemlose Hinzufügen, Löschen und Bearbeiten von Benutzerkonten. |
| `version` | | Zeigt die Versionsinformationen des Shelf Managers an. |

Die meisten Informationsbefehle unterstützen kurze und ausführliche Ausführungsmodi, die sich im bereitgestellten Informationsvolumen unterscheiden. Der kurze Modus ist der Standardmodus. Der ausführliche Modus wird ausgewählt, indem die Option `-v` in der Befehlszeile direkt nach dem Befehl und vor den Positionsargumenten verwendet wird.

# Überwachen des Systems

Die CLI von Shelf Manager stellt viele Befehle für die Überwachung des Systems und für das Anzeigen des Systemstatus bereit. In diesem Abschnitt werden verschiedene Möglichkeiten für das Überwachen Ihres Systems beschrieben. Weitere Informationen finden Sie unter „CLI-Befehle“ auf Seite 66 oder im Anhang A unter „Shelf Manager-CLI-Befehle“ auf Seite 125.

## Anzeigen von Platinen- und IPMC-Informationen

Platineninformationen umfassen Informationen über jeden IPM-Controller im Bereich der IPMB-Adressen, die ATCA-Steckplätzen zugewiesen sind, sowie über jede weitere FRU, die von diesen Controllern gesteuert wird. Der Bereich von IPMB-Adressen umfasst 82h-A0h für PICMG 3.0-Systeme, bei denen Platinen IPM-Controller enthalten.

Beispiele der folgenden Aufgaben werden zusammen mit den verwendeten Befehlen und den entsprechenden Ausgaben aufgeführt.

- So zeigen Sie Standardinformationen über alle Platinen im Server an
- So zeigen Sie detaillierte Informationen über eine Platine an
- So listen Sie die Sensoren auf einer Platine auf

- So zeigen Sie Daten von einem Sensoren auf einer Platine an
- So listen Sie alle IPMCs im Server auf
- So zeigen Sie Informationen über einen spezifischen IPM-Controller an
- So zeigen Sie detaillierte Informationen über den IPM-Controller an

## So zeigen Sie Standardinformationen über alle Platinen im Server an

In diesem Beispiel sind nur die Platinen in den physikalischen Steckplätzen 1 und 14 vorhanden.

```
# clia board
IPM Sentry Shelf Manager Command Line Interpreter
Physical Slot # 1
82: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
82: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
Physical Slot # 14
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
#
```

## So zeigen Sie detaillierte Informationen über eine Platine an

In diesem Beispiel werden detaillierte Informationen über die Platine im physikalischen Steckplatz 14 angezeigt.

```
# clia board -v 14
Physical Slot # 14
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID: 0x00, Revision: 0, Firmware: 1.01, IPMI ver 1.5
    Manufacturer ID: 00315a (PICMG), Product ID: 0000, Auxiliary
Rev: 01ac1014
```

```
    Device ID String: "IPM Sentry 6"
    Global Initialization: 0x0, Power State Notification: 0x0,
Device Capabilities: 0x29
    Controller provides Device SDRs
    Supported features: 0x29
        "Sensor Device" "FRU Inventory Device" "IPMB Event
Generator"
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
    Site Type: 0x00, Site Number: 14
    Current Power Level: 0x01, Maximum Power Level: 0x01, Current
Power Consumption: 20.0 Watts
#
```

- **So listen Sie die Sensoren auf einer Platine auf**

  In diesem Beispiel wird eine Liste der Sensoren auf der Platine unter der IPMB-Adresse 92 angezeigt.

```
# clia sensor 92
IPM Sentry Shelf Manager Command Line Interpreter

92: LUN: 0, Sensor # 0 ("Hot Swap")
    Type: Discrete (0x6f), "Hot Swap" (0xf0)
    Belongs to entity: (0xa0, 96) [FRU # 0]
92: LUN: 0, Sensor # 1 ("IPMB Physical")
    Type: Discrete (0x6f), "IPMB Link" (0xf1)
    Belongs to entity: (0xa0, 96) [FRU # 0]

92: LUN: 0, Sensor # 255 ("BMC Watchdog")
    Type: Discrete (0x6f), "Watchdog 2" (0x23)
    Belongs to entity: (0x3, 96) [FRU # 0]

92: LUN: 0, Sensor # 255 ("+12.0V")
    Type: Threshold (0x01), "Voltage" (0x02)
    Belongs to entity: (0x14, 96) [FRU # 0]

92: LUN: 0, Sensor # 255 ("+5.0V")
    Type: Threshold (0x01), "Voltage" (0x02)
    Belongs to entity: (0x14, 96) [FRU # 0]

92: LUN: 0, Sensor # 255 ("+3.3V")
    Type: Threshold (0x01), "Voltage" (0x02)
    Belongs to entity: (0x14, 96) [FRU # 0]
```

```
92: LUN: 0, Sensor # 255 ("+2.5V")
    Type: Threshold (0x01), "Voltage" (0x02)
    Belongs to entity: (0x14, 96) [FRU # 0]

92: LUN: 0, Sensor # 2 ("CPU1 Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Belongs to entity: (0x3, 96) [FRU # 0]

92: LUN: 0, Sensor # 3 ("CPU2 Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Belongs to entity: (0x3, 96) [FRU # 0]

92: LUN: 0, Sensor # 4 ("Inlet Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Belongs to entity: (0x3, 96) [FRU # 0]
    Type: Threshold (0x01), "Temperature" (0x01)
    Belongs to entity: (0x3, 96) [FRU # 0]
#
```

- **So zeigen Sie Daten von einem Sensoren auf einer Platine an**

In diesem Beispiel werden Informationen über Sensornummer 3 (CPU2 Temp) auf der Platine unter der IPMB-Adresse `92` angezeigt.

```
# clia sensor 92 3
IPM Sentry Shelf Manager Command Line Interpreter

92: LUN: 0, Sensor # 3 ("CPU2 Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Belongs to entity: (0x3, 96) [FRU # 0]
```

- **So listen Sie alle IPMCs im Server auf**

In diesem Beispiel wird die typische Ausgabe für den Befehl `ipmc` gezeigt.

```
# clia ipmc
IPM Sentry Shelf Manager Command Line Interpreter
10: Entity: (0xf0, 0x60) Maximum FRU device ID: 0x08
    PICMG Version 2.1
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)

20: Entity: (0xf0, 0x1) Maximum FRU device ID: 0x08
    PICMG Version 2.1
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
```

```
82: Entity: (0xa0, 0x60) Maximum FRU device ID: 0x00
    PICMG Version 2.1
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)

88: Entity: (0xa0, 0x60) Maximum FRU device ID: 0x00
   Hot Swap State: M7 (Communication Lost), Previous: M4 (Active),
Last State Change Cause: Communication Lost (0x4)

92: Entity: (0xa0, 0x60) Maximum FRU device ID: 0x00
    PICMG Version 2.1
   Hot Swap State: M4 (Active), Previous: M7 (Communication Lost),
Last State Change Cause: Communication Lost (0x4)

96: Entity: (0xa0, 0x60) Maximum FRU device ID: 0x00
    Hot Swap State: M7 (Communication Lost), Previous: M6
(Deactivation In Progress), Last State Change Cause: Communication
Lost (0x4)
20: Entity: (0xf0, 0x1) Maximum FRU device ID: 0x08
    PICMG Version 2.1
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
```

- **So zeigen Sie Informationen über einen spezifischen IPM-Controller an**

In diesem Beispiel werden allgemeine Informationen über den IPM-Controller bei der Adresse 9C angezeigt.

```
# clia ipmc 9c
IPM Sentry Shelf Manager Command Line Interpreter
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
#
```

- **So zeigen Sie detaillierte Informationen über den IPM-Controller an**

In diesem Beispiel werden detaillierte Informationen über den IPM-Controller bei der Adresse 9C angezeigt.

```
# clia ipmc -v 9c
IPM Sentry Shelf Manager Command Line Interpreter
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
```

```
    Device ID: 0x00, Revision: 0, Firmware: 1.01, IPMI ver 1.5
    Manufacturer ID: 00315a (PICMG), Product ID: 0000, Auxiliary
Rev: 01ac10ac
    Device ID String: "IPM Sentry 6"
    Global Initialization: 0x0, Power State Notification: 0x0,
Device Capabilities: 0x29
    Controller provides Device SDRs
    Supported features: 0x29
        "Sensor Device" "FRU Inventory Device" "IPMB Event
Generator"
#
```

# Anzeigen von FRU-Informationen

Sie können Informationen über alle FRUs im System anzeigen, indem Sie den Befehl `clia fru` ohne Parameter eingeben. Sie können auch Informationen über eine bestimmte FRU anzeigen, indem Sie die Adresse der FRU und optional die FRU-ID angeben. Informationen zum Zuordnen einer Chassis-Steckplatznummer zu einer physikalischen Adresse zu einer IPMB-Adresse finden Sie unter „Zuordnung von physikalischen Adressen zu logischen Steckplätzen“ auf Seite 14.

Im folgenden Beispiel in der Zeile „`20: FRU # 1`“ ist `20` die Mittelplatine der IPMB-Adresse, und `1` ist die FRU-Geräte-ID.

```
# clia fru
20: FRU # 1
 Entity: (0xf2, 0x60)
Hot Swap State: M4 (Active), Previous: M3 (Activation In Process),
Last State Change Cause: Normal State Change (0x0)
 Device ID String: "Shelf EEPROM 1"
```

## Anordnung der FRU-Informationen

In ABBILDUNG 3-1 wird dargestellt, wie die IPMI-FRU-Informationen angeordnet sind. Jede Partition stellt spezifische Datentypen bereit.

- Der „Common Header“ (Gemeinsamer Header) enthält Bereichs-Offsets.
- Der Bereich „Internal Use“ (Interne Zwecke) wird zum Speichern von proprietären Daten verwendet.
- Der Bereich „Chassis Information“ (Gehäuse-Informationen) enthält den Chassistyp, die Teilenummer und die Seriennummer.
- Der Bereich „Board Information“ (Board-Informationen) enthält den Hersteller-Zeitstempel, den Hersteller, den Produktnamen und die Teile-/Seriennummer.

- Der Bereich „Product Information“ (Produktinformationen) enthält den Hersteller, den Produktnamen, die Teile-/Seriennummer und die Version.
- Der Bereich „MultiRecord“ (Mehrfachdatensatz-Informationen) enthält dynamische Daten.

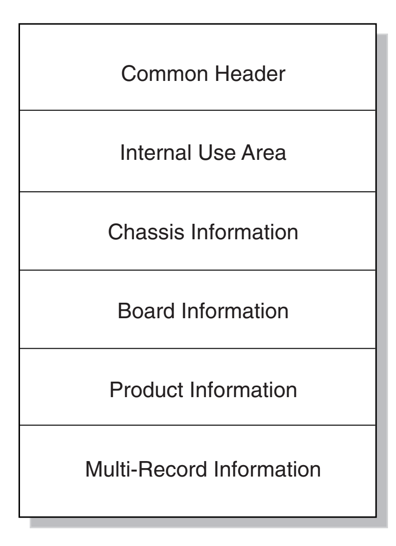



**ABBILDUNG 3-1** Anordnung der IPMI-FRU-Informationen

## Umgebung-FRUs

Umgebung-FRUs beinhalten Mittelplatine, Shelf-Verwaltungskarten, Lüfter, PEMs (Power Entry Modules) und das SAP (Shelf Alarm Panel). Alle Umgebung-FRUs haben nur IPMI FRU-Informationen, die von Drittanbietern bereitgestellt und programmiert werden.

Die Mittelplatinen-FRU-Informationen enthalten die Sun-Teilenummer sowie die Teilenummer des Drittanbieters. Sun fügt zusätzliche Informationen, wie z. B. Steckplatz, VLAN, VTAG und andere Daten, im Bereich „MultiRecord“ der FRU-Informationen der Mittelplatine hinzu. Die FRU-Informationen der Mittelplatine werden in zwei identischen EEPROMs gespeichert. Jede Änderung in einem EEPROM wird automatisch in dem anderen EEPROM durchgeführt.

## Blade-FRUs

Die Netra CT 900 Server-Hubboards (in den Steckplätzen 7 und 8) verfügen nur über IPMI-FRU-Informationen. Die Node Boards von Sun verfügen über zwei separate EEPROMS, von denen einer IPMI-FRU-Informationen und der andere Sun-FRU-Informationen enthält.

## Beispiele

Beispiele der folgenden Aufgaben werden zusammen mit den verwendeten Befehlen und den entsprechenden Ausgaben aufgeführt.

- So zeigen Sie Standardinformationen über alle FRUs im Shelf an
- So zeigen Sie Standardinformationen über alle FRUs bei der Adresse 9C an
- So zeigen Sie detaillierte Informationen über FRU 1 bei der Adresse 20 an
- So zeigen Sie FRU-Informationen in Rohform an
- So zeigen Sie FRU-Informationen im benutzerfreundlichen Format an

### ● So zeigen Sie Standardinformationen über alle FRUs im Shelf an

In diesem Beispiel werden die standardmäßigen FRU-Informationen dargestellt, die mit dem Befehl `fru` ohne Argumente angezeigt werden.

```
# clia fru
IPM Sentry Shelf Manager Command Line Interpreter

10: FRU # 0
    Entity: (0xf0, 0x60)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "ShMM-500"

12: FRU # 0
    Entity: (0xf0, 0x60)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "ShMM-500"

20: FRU # 0
    Entity: (0xf0, 0x1)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry BMC"

20: FRU # 1
    Entity: (0xf2, 0x60)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "Shelf EEPROM 1"

20: FRU # 2
    Entity: (0xf2, 0x61)
```

```
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: “Shelf EEPROM 2”

20: FRU # 3
    Entity: (0x7, 0x6f)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: “SAP Board”

20: FRU # 4
    Entity: (0x1e, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: “Fan Tray 0”

20: FRU # 5
    Entity: (0x1e, 0x1)
    Device ID String: “Fan Tray 1”revious: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)

20: FRU # 6
    Entity: (0x1e, 0x2)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: “Fan Tray 2”

20: FRU # 7
    Entity: (0xa, 0x60)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: “PEM A”

20: FRU # 8
    Entity: (0xa, 0x61)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: “PEM B”

82: FRU # 0
    Entity: (0xa0, 0x60)
   Hot Swap State: M7 (Communication Lost), Previous: M4 (Active),
Last State Change Cause: Unknown (0xf)
    Device ID String: “ATS1460”
```

```
9a: FRU # 0
    Entity: (0xa0, 0x60)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: “NetraCP-3010”
```

## So zeigen Sie Standardinformationen über alle FRUs bei der Adresse 9C an

In diesem Beispiel werden nur FRU-Informationen für alle FRUs bei der physikalischen Adresse 9C angezeigt.

```
# clia fru 9c
IPM Sentry Shelf Manager Command Line Interpreter
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
#
```

## So zeigen Sie detaillierte Informationen über FRU 1 bei der Adresse 20 an

In diesem Beispiel werden detaillierte FRU-Informationen für die FRU-Geräte-ID 1 bei der physikalischen Adresse 20 angezeigt.

```
# clia fruinfo 20 1
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1, FRU Info
Common Header:    Format Version = 1

Internal Use Area:
    Version = 1
Chassis Info Area:
    Version   = 1
    Chassis Type            = (23)
    Chassis Part Number     = 11592-450
    Chassis Serial Number   =

Board Info Area:
    Version     = 1
    Language Code           = 25
   Mfg Date/Time          = Jun 16 00:00:00 2005 (4973760 minutes
                            since 1996)
    Board Manufacturer      = Schroff
```

```
   Board Product Name       = ShMM-ACB-III Shelf Manager (Radial
                              IPMB)
    Board Serial Number      = 0000001
    Board Part Number        = 21593-251
    FRU Programmer File ID   = Schroff_11592450_AA.inf

Product Info Area:
    Version     = 1
    Language Code            = 25
    Manufacturer Name        = Schroff
    Product Name             = 12U 14-Slot ATCA Chassis
    Product Part / Model#    = 11592-450
    Product Version          = Dual Star (Radial IPMB)
    Product Serial Number    = 0000001
    Asset Tag                =
    FRU Programmer File ID   = Schroff_11592450_AA.inf

Multi Record Area:
    PICMG Shelf Manager IP Connection Record (ID=0x13)
        Version = 1

    Record Type              = Management Access Record
        Version = 2
    Sub-Record Type: Component Name (0x05)

    PICMG Address Table Record (ID=0x10)
        Version = 0

    PICMG Backplane Point-to-Point Connectivity Record (ID=0x04)
        Version = 0

    PICMG Backplane Point-to-Point Connectivity Record (ID=0x04)
        Version = 0

    PICMG Shelf Activation And Power Management Record (ID=0x12)
        Version = 0

    PICMG Shelf Power Distribution Record (ID=0x11)
        Version = 0

    PICMG Radial IPMB-0 Link Mapping Record (ID=0x15)
        Version = 0

    Record Type              = 0xf0 OEM Record
        Version = 2
```

## ● So zeigen Sie FRU-Informationen in Rohform an

In diesem Beispiel werden FRU-Informationen für alle FRUs und für eine spezifische FRU in Rohform angezeigt.

```
# clia frudata
IPM Sentry Shelf Manager Command Line Interpreter
10: FRU # 0 Raw FRU Info Data
    FRU Info size: 435
12: FRU # 0 Raw FRU Info Data
    FRU Info size: 435
20: FRU # 0 Raw FRU Info Data
    FRU Info size: 152
20: FRU # 1 Raw FRU Info Data
    FRU Info size: 8192
20: FRU # 2 Raw FRU Info Data
    FRU Info size: 8192
20: FRU # 3 Raw FRU Info Data
    FRU Info size: 2048
20: FRU # 4 Raw FRU Info Data
    FRU Info size: 2048
20: FRU # 5 Raw FRU Info Data
    FRU Info size: 2048
20: FRU # 6 Raw FRU Info Data
    FRU Info size: 2048
20: FRU # 7 Raw FRU Info Data
    FRU Info size: 2048
20: FRU # 8 Raw FRU Info Data
    FRU Info size: 2048
20: FRU # 254 Raw FRU Info Data
    FRU Info size: 3068
9a: FRU # 0 Raw FRU Info Data
    FRU Info size: 512
#
```

```
# clia frudata 20 1 0
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1 Block # 0 Raw FRU Info Data
    FRU Info size: 8192
 01 01 22 24  31 3E 00 49  01 A0 A1 A2  A3 A4 A5 A6
 A7 A8 A9 AA  AB AC AD E0  E1 E2 E3 E4  E5 E6 D0 D1
```

- **So zeigen Sie FRU-Informationen im benutzerfreundlichen Format an**

In diesem Beispiel wird eine benutzerfreundliche Version der FRU-Informationen gezeigt.

```
# clia fruinfo 20 1
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1, FRU Info
Common Header:    Format Version = 1

Internal Use Area:
    Version = 1
Chassis Info Area:
    Version   = 1
    Chassis Type             = (23)
    Chassis Part Number      = 11592-450
    Chassis Serial Number    =

Board Info Area:
    Version     = 1
    Language Code            = 25
   Mfg Date/Time          = Jun 16 00:00:00 2005 (4973760 minutes
                              since 1996)
    Board Manufacturer       = Schroff
    Board Product Name       = ShMM-ACB-III Shelf Manager (Radial
                              IPMB)
    Board Serial Number      = 0000001
    Board Part Number        = 21593-251
    FRU Programmer File ID   = Schroff_11592450_AA.inf

Product Info Area:
    Version     = 1
    Language Code            = 25
    Manufacturer Name        = Schroff
    Product Name             = 12U 14-Slot ATCA Chassis
    Product Part / Model#    = 11592-450
    Product Version          = Dual Star (Radial IPMB)
    Product Serial Number    = 0000001
    Asset Tag                =
    FRU Programmer File ID   = Schroff_11592450_AA.inf
```

```
Multi Record Area:
    PICMG Shelf Manager IP Connection Record (ID=0x13)
        Version = 1

    Record Type              = Management Access Record
        Version = 2
    Sub-Record Type: Component Name (0x05)

    PICMG Address Table Record (ID=0x10)
        Version = 0

    PICMG Backplane Point-to-Point Connectivity Record (ID=0x04)
        Version = 0

    PICMG Backplane Point-to-Point Connectivity Record (ID=0x04)
        Version = 0

    PICMG Shelf Activation And Power Management Record (ID=0x12)
        Version = 0

    PICMG Shelf Power Distribution Record (ID=0x11)
        Version = 0

    PICMG Radial IPMB-0 Link Mapping Record (ID=0x15)
        Version = 0

    Record Type              = 0xf0 OEM Record
        Version = 2
    UNKNOWN Manufacturer ID = 0x303833
```

# Anzeigen von Shelf-Informationen

Sie können den Befehl `clia shelf` mit einem der gültigen Parameter verwenden, um FRU-Informationen für wichtige Shelf-FRUs anzuzeigen. Sie können außerdem aktuelle Betriebsdaten für das Shelf abrufen und einige Felder in den Shelf-FRU-Informationen bearbeiten. Die gültigen `Shelf`-Parameter lauten:

- `cooling_state` oder `cs`
- `fans_state` oder `fs`
- `address_table` oder `at`
- `power_distribution` oder `pd`
- `power_management` oder `pm`

- pci_connectivity oder pcic
- ha_connectivity oder ha
- h110_connectivity oder h1110c
- point-to-point_connectivity oder ppc

Weitere Informationen finden Sie unter „Anzeigen von Shelf-FRU-Informationen“ auf Seite 238.

## Beispiele

Beispiele der folgenden Aufgaben werden zusammen mit den verwendeten Befehlen und den entsprechenden Ausgaben aufgeführt.

- So zeigen Sie den Shelf-Kühlstatus an
- So zeigen Sie den Shelf-Lüfterstatus an
- So zeigen Sie die Adresstabelle an
- So zeigen Sie Energieverwaltungsinformationen an
- So zeigen Sie Energieverteilungsinformationen an

### ● So zeigen Sie den Shelf-Kühlstatus an

In diesem Beispiel werden die Befehle und Ausgaben für das Anzeigen des Shelf-Kühlstatus gezeigt.

```
# clia shelf cooling_state
IPM Sentry Shelf Manager Command Line Interpreter
    Cooling state: "Normal"

# clia shelf -v cooling_state
IPM Sentry Shelf Manager Command Line Interpreter

    Cooling state: "Normal"
    Sensor(s) at this state: (0x9a,4,0) (0x9a,5,0) (0x10,2,0) (0x9a,3,0)
                           (0x20,120,0) (0x20,121,0) (0x20,122,0) (0x20,123,0)
                           (0x20,200,0) (0x20,201,0) (0x20,240,0) (0x20,241,0)
                              (0x20,242,0)
```

### ● So zeigen Sie den Shelf-Lüfterstatus an

Die Befehle zum Anzeigen des Lüfterstatus und die Ausgaben werden in diesem Beispiel gezeigt.

```
# clia shelf fans_state
IPM Sentry Shelf Manager Command Line Interpreter
    Fans state: "Normal"
```

```
# clia shelf -v fans_state
IPM Sentry Shelf Manager Command Line Interpreter
    Fans state: "Normal"
    Sensor(s) at this state: (0x10,7,0) (0x10,8,0) (0x10,9,0) (0x10,10,0)
                             (0x10,11,0) (0x10,12,0)
#
```

- **So zeigen Sie die Adresstabelle an**

  In diesem Beispiel werden die Befehle und Ausgaben für das Anzeigen der Shelf-Adresstabelle gezeigt.

```
# clia shelf address_table

IPM Sentry Shelf Manager Command Line Interpreter

    PICMG Address Table Record (ID=0x10)
        Version = 0
    Shelf Address            = 1
    Address Table Entries#   = 16
        Hw Addr: 41, Site # 7, Type: "AdvancedTCA Board" 00
        Hw Addr: 42, Site # 8, Type: "AdvancedTCA Board" 00
        Hw Addr: 43, Site # 6, Type: "AdvancedTCA Board" 00
        Hw Addr: 44, Site # 9, Type: "AdvancedTCA Board" 00
        Hw Addr: 45, Site # 5, Type: "AdvancedTCA Board" 00
        Hw Addr: 46, Site # 10, Type: "AdvancedTCA Board" 00
        Hw Addr: 47, Site # 4, Type: "AdvancedTCA Board" 00
        Hw Addr: 48, Site # 11, Type: "AdvancedTCA Board" 00
        Hw Addr: 49, Site # 3, Type: "AdvancedTCA Board" 00
        Hw Addr: 4a, Site # 12, Type: "AdvancedTCA Board" 00
        Hw Addr: 4b, Site # 2, Type: "AdvancedTCA Board" 00
        Hw Addr: 4c, Site # 13, Type: "AdvancedTCA Board" 00
        Hw Addr: 4d, Site # 1, Type: "AdvancedTCA Board" 00
        Hw Addr: 4e, Site # 14, Type: "AdvancedTCA Board" 00
        Hw Addr: 08, Site # 1, Type: "Dedicated ShMC" 03
        Hw Addr: 09, Site # 2, Type: "Dedicated ShMC" 03
```

```
# clia shelf fans_state
IPM Sentry Shelf Manager Command Line Interpreter
    Fans state: "Normal"

# clia shelf -v fans_state
IPM Sentry Shelf Manager Command Line Interpreter
    Fans state: "Normal"
    Sensor(s) at this state: (0x10,7,0) (0x10,8,0) (0x10,9,0) (0x10,10,0)
                             (0x10,11,0) (0x10,12,0)
#
```

- **So zeigen Sie Energieverwaltungsinformationen an**

  Ein Beispiel des Befehls und der entsprechenden Ausgabe werden in diesem Beispiel gezeigt.

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
        Version = 0
    Allowance for FRU Activation Readiness: 20 seconds
    FRU Activation and Power Description Count: 19
    Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 42, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 43, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 44, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 45, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Wattss
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 46, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 47, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 48, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 49, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 4a, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
```

```
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 4b, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 4c, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 4d, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 4e, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 44, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 45, FRU ID: 0xfe, Maximum FRU Power Capabilities: 200 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 08, FRU ID: 0xfe, Maximum FRU Power Capabilities: 24 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 09, FRU ID: 0xfe, Maximum FRU Power Capabilities: 24 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 20, FRU ID: 0xfe, Maximum FRU Power Capabilities: 100 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

#
```

- **So zeigen Sie Energieverteilungsinformationen an**

In diesem Beispiel werden die Befehle und Ausgabe für das Abrufen der Energieverteilungsinformationen für das Shelf gezeigt.

```
# clia shelf pd
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Power Distribution Record (ID=0x11)
        Version = 0
```

```
Feed count: 8
Feed 00:
    Maximum External Available Current: 25.0 Amps
    Maximum Internal Current: 25.0 Amps
    Minimum Expected Operating Voltage: -40.5 Volts
    Actual Power Available: 1012.500 Watts
    Currently Used Power: 30.000 Watts
    Feed-to-FRU Mapping entries count: 3
       FRU Addr: 49, FRU ID: 0xfe
       FRU Addr: 4b, FRU ID: 0xfe
       FRU Addr: 4d, FRU ID: 0xfe
Feed 01:
    Maximum External Available Current: 25.0 Amps
    Maximum Internal Current: 25.0 Amps
    Minimum Expected Operating Voltage: -40.5 Volts
    Actual Power Available: 1012.500 Watts
    Currently Used Power: 30.000 Watts
    Feed-to-FRU Mapping entries count: 3
       FRU Addr: 49, FRU ID: 0xfee
       FRU Addr: 4b, FRU ID: 0xfe
       FRU Addr: 4d, FRU ID: 0xfe
Feed 02:
    Maximum External Available Current: 25.0 Amps
    Maximum Internal Current: 25.0 Amps
    Minimum Expected Operating Voltage: -40.5 Volts
    Actual Power Available: 1012.500 Watts
    Currently Used Power: 40.000 Watts
    Feed-to-FRU Mapping entries count: 4
       FRU Addr: 41, FRU ID: 0xfe
       FRU Addr: 43, FRU ID: 0xfe
       FRU Addr: 45, FRU ID: 0xfe
       FRU Addr: 47, FRU ID: 0xfe
Feed 03:
    Maximum External Available Current: 25.0 Amps
    Maximum Internal Current: 25.0 Amps
    Minimum Expected Operating Voltage: -40.5 Volts
    Actual Power Available: 1012.500 Watts
    Currently Used Power: 40.000 Watts
    Feed-to-FRU Mapping entries count: 4
       FRU Addr: 41, FRU ID: 0xfe
       FRU Addr: 43, FRU ID: 0xfe
       FRU Addr: 45, FRU ID: 0xfe
       FRU Addr: 47, FRU ID: 0xfe
    Maximum External Available Current: 25.0 Amps
    Maximum Internal Current: 25.0 Amps
```

```
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 1012.500 Watts
        Currently Used Power: 40.000 Watts
        Feed-to-FRU Mapping entries count: 4
           FRU Addr: 42, FRU ID: 0xfe
           FRU Addr: 44, FRU ID: 0xfe
           FRU Addr: 46, FRU ID: 0xfe
           FRU Addr: 48, FRU ID: 0xfe
    Feed 05:
        Maximum External Available Current: 25.0 Amps
        Maximum Internal Current: 25.0 Amps
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 1012.500 Watts
        Currently Used Power: 40.000 Watts
        Feed-to-FRU Mapping entries count: 4
           FRU Addr: 42, FRU ID: 0xfe
           FRU Addr: 44, FRU ID: 0xfe
           FRU Addr: 46, FRU ID: 0xfe
           FRU Addr: 48, FRU ID: 0xfe
    Feed 06:
        Maximum External Available Current: 25.0 Amps
        Maximum Internal Current: 25.0 Amps
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 1012.500 Watts
        Currently Used Power: 100.000 Watts
        Feed-to-FRU Mapping entries count: 6
           FRU Addr: 08, FRU ID: 0xfe
           FRU Addr: 09, FRU ID: 0xfe
           FRU Addr: 20, FRU ID: 0xfe
           FRU Addr: 4a, FRU ID: 0xfe
           FRU Addr: 4c, FRU ID: 0xfe
           FRU Addr: 4e, FRU ID: 0xfe
    Feed 07:
        Maximum External Available Current: 25.0 Amps
        Maximum Internal Current: 25.0 Amps
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 1012.500 Watts
        Currently Used Power: 100.000 Watts
        Feed-to-FRU Mapping entries count: 6
           FRU Addr: 08, FRU ID: 0xfe
           FRU Addr: 09, FRU ID: 0xfe
           FRU Addr: 20, FRU ID: 0xfe
           FRU Addr: 4a, FRU ID: 0xfe
```

```
            FRU Addr: 4c, FRU ID: 0xfe
            FRU Addr: 4e, FRU ID: 0xfe
#
```

# Neuinitialisieren des Shelf Managers

In diesem Abschnitt wird beschrieben wie die U-Boot-Umgebungsvariablen, das Dateisystem im Flash-Speicher und das Anmeldepasswort auf der Shelf-Verwaltungskarte-Karte neu initialisiert wird.

## Neuinitialisieren der U-Boot-Umgebung

Die U-Boot-Umgebungsvariablen werden im EEPROM der Shelf-Verwaltungskarte gespeichert. Zum Wiederherstellen der werkseitigen Einstellungen für die U-Boot-Umgebungsvariablen müssen Sie zuerst die im EEPROM gespeicherten Umgebungsvariablen löschen und die Shelf-Verwaltungskarte zurücksetzen (oder das System neu starten).

### ▼ So initialisieren Sie die U-Boot-Umgebung neu

1. **Löschen Sie den EEPROM, indem Sie den folgenden Befehl in der U-Boot-Eingabeaufforderung eingeben:**

```
ShMM # eeprom write 80400000 0 1000

EEPROM @0x50 write: addr 80400000  off 0000  count 4096 ... done
ShMM #
```

2. **Setzen Sie die Shelf-Verwaltungskarte wie folgt zurück:**

```
ShMM # reset
U-Boot 1.1.2 (Nov 27 2005 - 19:17:09)

CPU: Au1550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 8000041
DRAM:  128 MB
Flash: 64 MB
*** Warning - bad CRC, using default environment

In:    serial
Out:   serial
Err:   serial
Net:   Au1X00 ETHERNET
Hit any key to stop autoboot:  0

ShMM #
```

3. **Speichern Sie diese Umgebungseinstellungen mithilfe des folgenden Befehls:**

```
ShMM # saveenv
```

# Neuinitialisieren des Dateisystems

Das Dateisystem ist im Flash-Speicher gespeichert und kann relativ einfach auf die werkseitigen Einstellungen zurückgesetzt werden. U-Boot verfügt über eine Umgebungsvariable mit dem Namen `flash_reset`. Wenn Sie für diese Variable `y` festlegen und anschließend das System starten, wird das Dateisystem auf die werkseitigen Einstellungen zurückgesetzt.

```
ShMM # setenv flash_reset y
ShMM # boot
```

Für die Variable `flash_reset` wird beim Starten des Systems automatisch `n` festgelegt, nachdem der Flash-Speicher neu initalisisiert wurde. Mit dem Befehl `bootcmd` wird der Linux-Kernel gestartet. Während dieses Vorgangs wird das Dateisystem neu initialisiert. Die folgende Ausgabe wird auf der Konsole angezeigt.

```
/etc/rc: Mounted /dev/pts
/etc/rc: Flash erase requested via U-BOOT var
/etc/rc: erasing mtdchar1 -> /etc
Erased 1024 Kibyte @ 0 -- 100% complete.
/etc/rc: erasing mtdchar0 -> /var
Erased 1536 Kibyte @ 0 -- 100% complete.
/etc/rc: Mounted /dev/mtdblock3 to /var
/etc/rc: /var/log mounted as FLASH disk
/etc/rc: Started syslogd and klogd
/etc/rc: /var/tmp mounted as RAM disk
/etc/rc: hostname demo
/etc/rc: /dev/mtdblock2 appears to be empty ... restoring from
factory /etc...
```

# Zurücksetzen des Anmeldepassworts

Werkseitig ist der Anmeldename für die Shelf-Verwaltungskarte eine Benutzer-ID von `root` ohne Passwort. Die Benutzer sollten das Passwort bei der Konfiguration von Shelf Manager ändern. Falls das neue Passwort vergessen wurde, kann es über die U-Boot-Variable `password_reset` auf die werkseitige Einstellung zurückgesetzt werden. Wenn Sie für diese Variable `y` festlegen und anschließend das System starten, wird das `root`-Passwort entfernt.

```
ShMM # setenv password_reset y
ShMM # boot
```

Die folgende Ausgabe wird während des Startens auf der Konsole angezeigt.

```
/etc/rc: hostname demo
```

# Umprogrammieren der Shelf-Verwaltungskarte

Um eine Shelf-Verwaltungskarte umzuprogrammieren, müssen mehrere darauf gespeicherte Firmware-Abbilder umprogrammiert werden. Diese Abbilder können abhängig von der für die Umprogrammierung verwendeten Methode in zwei Gruppen aufgeteilt werden. Diese Gruppen lauten:

- U-Boot-, Linux-Kernel- und Linux-RFS (Root File System)-Abbilder

  Diese Abbilder werden normalerweise mithilfe des zuverlässigen Aktualisierungsvorgangs umprogrammiert. Alternativ dazu kann der Kernel und RFS über die U-Boot-Eingabeaufforderung umprogrammiert werden, in dem sie von einem TFTP-Server geladen werden.

- CPLD (Complex Programmable Logic Device)-Abbild

  Dieses Abbild wird über die Befehlszeile der Shelf-Verwaltungskarte mithilfe eines speziellen Befehlszeilendienstprogramms namens `cpldtool` umprogrammiert.

Im Folgenden erhalten Sie detaillierte Anweisungen für das Umprogrammieren dieser Abbilder.

# Zuverlässiger Aktualisierungsvorgang für Firmware

Das Monterey-Linux-Betriebssystem bietet einen zuverlässigen Aktualisierungsvorgang für die Firmware-Abbilder auf einer ausgeführten und funktionierenden Shelf-Verwaltungskarte. Der Vorgang unterstützt die Aktualisierung der U-Boot-Firmware, des Linux-Kernel und des Linux-RFS (bzw. einer beliebigen Kombination aus diesen drei Abbildern). Falls ein Software-Aktualisierungsversuch fehlschlägt (z. B. aufgrund der Installation eines fehlerhaften U-Boot-Firmware-Abbilds, das die Shelf-Verwaltungskarte nicht starten kann, oder ein Shelf Manager, der nicht gestartet werden kann), greift der zuverlässige Aktualisierungsvorgang automatisch auf die vorherige Version der Firmware im persistenten Flash-Speicher zurück.

Der Flash-Speicher der Shelf-Verwaltungskarte ist in zwei Bereiche aufgeteilt. Wenn ein stabiler Satz von Firmware in einem dieser Bereiche eingerichtet ist, wird dieser als *persistenter* Bereich bezeichnet. Wenn neue Firmware installiert wird, geschieht dies im anderen Bereich, der anfänglich als *provisorisch* gekennzeichnet wird.

Nachdem ein neuer Satz Firmware im provisorischen Bereich überprüft wurde, wird dieser Bereich als der persistente Bereich festgelegt und wird verwendet, bis ein neuer Aktualisierungszyklus den Vorgang erneut startet.

Unabhängig davon, was im provisorischen Flash-Speicher installiert ist, stellen die zuverlässigen Hardwareaktualisierungsmechanismen sicher, dass es der Shelf-Verwaltungskarte immer möglich ist, von einer Softwarekopie zu starten, die entweder voll funktionsfähig ist oder ausreichend funktionsfähig, um festzustellen, dass ein Fehler in der Aktualisierungssitzung aufgetreten ist und entsprechende Korrekturmaßnahmen ergreift, um zur sicheren Softwarekopie im persistenten Flash-Speicher zurückzukehren.

Auf einer höheren Ebene werden die zuverlässigen Hardwareaktualisierungsmechanismen von einem Softwareprotokoll unterstützt, das auf der Protokollierung des Status der Aktualisierungssitzung in einer nicht flüchtigen Datei in `/var/upgrade/status` basiert (siehe „Statusdatei des zuverlässigen Aktualisierungsvorgangs“ auf Seite 97). Das Softwareprotokoll stellt sicher, dass die zuverlässige Aktualisierung erst beendet wird, wenn alle erforderlichen Aktionen, einschließlich der benutzerdefinierten *hook*-Skripts, die für eine bestimmte Anwendung benötigt werden können, erfolgreich abgeschlossen wurden.

Der in Monterey Linux implementierte zuverlässige Aktualisierungsvorgang ist neutral hinsichtlich der integrierten Anwendung, die auf der Shelf-Verwaltungskarte ausgeführt wird. Der Vorgang stellt eine ausreichende Menge an Eingriffspunkten („Hooks“) bereit, die es ermöglichen, dass eine bestimmte Anwendung auf der Shelf-Verwaltungskarte ausgeführt wird, damit sichergestellt wird, dass benutzerdefinierte Aktionen bei den entsprechenden Punkten der zuverlässigen Aktualisierung ausgeführt werden. Im letzten Teil dieses Abschnitts werden Vorrichtungen für die zuverlässige Aktualisierung der IPM Sentry Shelf Manager-Firmware beschrieben, die mithilfe dieser Eingriffspunkte implementiert wurden.

## Flash-Partitionierung

Die Shelf-Verwaltungskarte stellt einen Hardwaremechanismus bereit, der das Austauschen der unteren und oberen Hälfte des Flash-Speichers in der Systemspeicherzuordnung unter Kontrolle der Software ermöglicht, die auf dem MIPS-Prozessor ausgeführt wird. Diese Funktion wurde zur Unterstützung des zuverlässigen Aktualisierungsvorgangs für Software-Abbilder im Flash-Speicher implementiert. Der zuverlässige Software-Aktualisierungsvorgang geht davon aus, dass das Flash-Gerät zwei Kopien der Software in der unteren und in der oberen Hälfte des Flash-Speichers enthält. Alle Shelf-Verwaltungskarte sind mit der Partitionierung ausgestattet, wobei das Flash-Gerät in zwei gleiche Teile aufgeteilt ist, die beide je eine Kopie der Shelf Manager-Software enthalten sollen.

Die U-Boot-Umgebungsvariable `reliable_upgrade` (siehe „U-Boot-Umgebungsvariablen“ auf Seite 19) wird von den Linux-Ebenen verwendet, um zu bestimmen, ob der zuverlässige Aktualisierungsvorgang aktiviert ist. Für diese Variable muss `y`

festgelegt werden und sie muss zum Linux-Kernel in der Kernel-Parameterzeichenfolge bootargs übergeben werden. Die Linux-Flash-MTD-Ebene prüft den Parameter reliable_upgrade zum Initialisierungszeitpunkt der Flash-Partitionen und partitioniert abhängig vom Parameterwert (sowie der Größe des auf der Shelf-Verwaltungskarte installierten Flash-Geräts) das Flash-Gerät auf die geeignete Weise.

In diesem Abschnitt wird davon ausgegangen, dass die Shelf-Verwaltungskarte so konfiguriert ist, dass sie die zuverlässige Aktualisierung unterstützt und die zwei getrennten Flash-Bereiche enthält. TABELLE 3-3 enthält eine Übersicht über die Flash-Partitionen auf der Shelf-Verwaltungskarte in dieser Konfiguration (reliable_upgrade=y):

**TABELLE 3-3** Flash-Partitionen für 16 MB reliable_upgrade=y

| Offset in Flash (in MB) | Größe (in MB) | Geräte-Knoten | Geladen als (beim Starten) | Inhalt |
|---|---|---|---|---|
| 0 | 0.5 | /dev/mtdchar10, /dev/mtdblock10 | /var/upgrade | Die zweite Hälfte des JFFS2-Dateisystems /var/upgrade |
| 0.5 + (FLASHGRÖSSE – 16) / 2 | 1.5 | /dev/mtdchar5, /dev/mtdblock5 | Nicht geladen | Das andere /var-JFFS2-Dateisystem |
| FLASHGRÖSSE / 2 – 62 | 1 | /dev/mtdchar6, /dev/mtdblock6 | Nicht geladen | Das andere /etc-JFFS2-Dateisystem |
| FLASHGRÖSSE / 2 – 53 | 1 | /dev/mtdchar7 | Nicht geladen | Das andere Linux-Kernel-Abbild |
| FLASHGRÖSSE / 2 – 44 | 0.25 | /dev/mtdchar8 | Nicht geladen | Das andere U-Boot-Firmware-Abbild |
| FLASHGRÖSSE / 2 – 3.754.25 | 3.75 | /dev/mtdchar9 | Nicht geladen | Das andere Linux-RFS (Root File System)-Abbild |
| FLASHGRÖSSE / 28 | 0.5 | /dev/mtdchar10, /dev/mtdblock10 | /var/upgrade | Die erste Hälfte des JFFS2-Dateisystems /var/upgrade |
| FLASHGRÖSSE – 7.58.5 | 1 | /dev/mtdchar0, /dev/mtdblock0 | /var | Das /var-JFFS2-Dateisystem |
| FLASHGRÖSSE – 610 | 1 | /dev/mtdchar1, /dev/mtdblock1 | /etc | Das /etc-JFFS2-Dateisystem |

**TABELLE 3-3** Flash-Partitionen für 16 MB `reliable_upgrade=y` *(Fortsetzung)*

| | | | | |
|---|---|---|---|---|
| FLASHGRÖSSE – 5.11 | 1 | /dev/mtdchar2 | Nicht geladen | Das Linux-Kernel-Abbild |
| FLASHGRÖSSE – 412 | 0.25 | /dev/mtdchar3 | Nicht geladen | Das U-Boot-Firmware-Abbild |
| FLASHGRÖSSE – 3.712.255 | 3.75 | /dev/mtdchar4 | Nicht geladen | Das Linux-RFS (Root File System)-Abbild |

## Das /var/upgrade-Dateisystem

Wenn für `reliable_upgrade` `y` festgelegt ist, lädt Monterey Linux wie unter „Zuverlässiger Aktualisierungsvorgang für Firmware“ auf Seite 94 beschrieben eine 1-MB-Partition als ein JFFS2-Dateisystem unter `/var/upgrade`. Dieses Dateisystem wird als Host für die Statusdatei des zuverlässigen Aktualisierungsvorgangs verwendet (siehe „Statusdatei des zuverlässigen Aktualisierungsvorgangs“ auf Seite 97).

Beachten Sie, dass die `/var/upgrade`-JFFS2-Partition aus zwei nicht aufeinander folgenden Flash-Blöcken (jeweils 0,5 MB) besteht, von denen sich einer in der unteren Hälfte und der andere in der oberen Hälfte des Flash-Geräts befindet. Monterey Linux nutzt die Fähigkeit von Linux-MTD- und JFFS2-Ebenen, ein Dateisystem in nicht aufeinander folgenden Flash-Sektoren zu unterstützen, um auf diese Weise `/var/upgrade` zu implementieren.

Eine weitere Eigenschaft des JFFS2-Dateisystems, aufgrund deren `/var/upgrade` den Aufgaben des zuverlässigen Aktualisierungsvorgangs dient, besteht darin, dass die internen JFFS2-Strukturen keine Abhängigkeiten (wie verknüpfte Listen) auf Grundlage von Flash-Sektornummern oder absoluten Offsets im Flash-Speicher erstellen. Wenn ein Dateisystem in einer Partition geladen wird, durchsucht JFFS2 stattdessen alle Flash-Sektoren, aus denen die Partition besteht, und erstellt den logischen Inhalt eines Dateisystems in einer internen In-RAM-Darstellung neu. Diese Funktion stellt sicher, dass Linux unabhängig davon, von welcher Hälfte des Flash-Speichers ShMM gestartet wurde, `/var/upgrade` als ein JFFS2-Dateisystem laden und den vorherigen Inhalt des Dateisystems verwenden kann.

## Statusdatei des zuverlässigen Aktualisierungsvorgangs

Die Software für den zuverlässigen Aktualisierungsvorgang verwaltet den Status der letzten Aktualisierungssitzung in der Datei `/var/upgrade/status`, die sich in einem dedizierten Dateisystem (`/var/upgrade`) befindet, das von Linux unabhängig davon geladen wird, von welchem Flash-Speicher ShMM gestartet wurde. Falls die Datei vorhanden ist, enthält sie den Status einer Aktualisierungssitzung, die entweder momentan ausgeführt wird oder vor kurzem abgeschlossen wurde.

/var/upgrade/status ist eine ASCII-Datei, die einen oder mehrere neue zeilenbegrenzte Datensätze enthalten, von denen jeder den Status eines bestimmten Schritts im Aktualisierungsvorgang beschreibt. Eine Datensatzzeile hat das folgende Format:

<*Schritt*>: <*Status*>

wobei *Schritt* eine Ganzzahl im Bereich von 1 bis 14 (wobei Schritt 14 einer abgeschlossenen Aktualisierungssitzung entspricht) und *Status* eine von einem Benutzer lesbare Zeichenkette ist, die den Status des aktuellen Schritts der Aktualisierungssitzung beschreibt.

Die Statusdatei wird vom Dienstprogramm für die zuverlässige Aktualisierung verwendet (siehe „Dienstprogramm für die zuverlässige Aktualisierung“ auf Seite 98), um ein Softwareprotokoll in der Schicht über den Hardware-Mechanismen des zuverlässigen Aktualisierungsvorgangs zu verwalten, damit der Status des Aktualisierungsvorgangs bestimmt und korrekt fortgesetzt werden kann.

## Dienstprogramm für die zuverlässige Aktualisierung

Für das Ausführen des zuverlässigen Aktualisierungsvorgangs sowie für das Prüfen des Status der neuesten Aktualisierung steht ein spezielles Benutzerbereich-Dienstprogramm zur Verfügung.

Das Dienstprogramm kann nur vom Superuser (root)-Konto aufgerufen werden. Jeder Versuch, das Dienstprogramm von einem anderen als dem Superuser-Konto auszuführen, wird zurückgewiesen.

Im ersten Schritt bei der Ausführung prüft das Dienstprogramm, ob für die U-Boot-Umgebungsvariable reliable_upgrade (siehe „Flash-Partitionierung“ auf Seite 95), die von U-Boot zum Linux-Kernel in der Kernel-Parameterzeichenfolge übergeben wurde, y festgelegt ist. Ist dies nicht der Fall, wird das Dienstprogramm sofort mit einem entsprechenden Fehlercode beendet.

Falls das Dienstprogramm mit der Option -s, -c oder -f aufgerufen wird, wird es verwendet, um den zuverlässigen Aktualisierungsvorgang auszuführen. Während des Aktualisierungsvorgangs protokolliert das Dienstprogramm in der Datei /var/upgrade/status den Status jeder Aktion, die es im Verlauf der einzelnen Schritte des Aktualisierungsvorgangs durchführt. Wenn das Dienstprogramm einen Fehler feststellt, wird der zuverlässige Aktualisierungsvorgang beendet, indem unter /var/upgrade/status ein Datensatz hinzugefügt wird, der auf einen nicht erfolgreichen Abschluss des Aktualisierungsvorgangs hinweist, und das Dienstprogramm wird mit einem entsprechenden Fehlercode beendet.

Das Dienstprogramm druckt alle Informationsmeldungen in stdout. Durch Hinzufügen des Bezeichners -v zu einer Option, die dies unterstützt, wird die Ausführlichkeit der Informationsmeldungen erhöht. Das Dienstprogramm druckt alle Fehlermeldungen in stderr.

Das Dienstprogramm hat die folgende Syntax:

- `rupgrade_tool -s {--dst=src}... [--proto=protocol] [-d] [--hook=args] [-v]`
- `rupgrade_tool -c [-v]`
- `rupgrade_tool -f [--hook=args] [-v]`
- `rupgrade_tool -w [-f]`
- `rupgrade_tool -S [-v]`
- `rupgrade_tool -u`
- `rupgrade_tool -h`

Die Parameter sind wie folgt definiert:

`-s {--dst=src}... [--proto=protocol] [--hook=args] [-v]`

Inititieren Sie den zuverlässigen Aktualisierungsvorgang. Dieser Schritt ist Teil der Unterstützung von Shelf Manager und umfasst die folgenden Aktionen:

- Erhalten der zu kopierenden Abbilder: lokal oder über das Netzwerk
- Kopieren der Abbilder in den provisorischen Flash-Speicher
- Beenden der Shelf Manager-Instanz, die auf dem ShMM ausgeführt wird (falls zutreffend)
- Kopieren der nicht flüchtigen Abbilder in den provisorischen Flash-Speicher
- Zurücksetzen der Shelf-Verwaltungskarte und Anweisen, dass sie vom provisorischen Flash-Speicher gestartet wird

  Aufgrund des letzten Schritts gibt ein Aufrufen von `rupgrade_tool -s` nichts zurück, sondern resultiert im Zurücksetzen der Shelf-Verwaltungskarte. Wenn `rupgrade_tool -s` etwas zurückgibt, wird damit angegeben, dass der zuverlässige Aktualisierungsvorgang fehlgeschlagen ist und beendet wurde, bevor mit dem Zurücksetzen der Shelf-Verwaltungskarte fortgefahren wurde, um sie vom provisorischen Flash-Speicher zu starten.

  Bevor der erste Schritt des Aktualisierungsvorgangs durch das Dienstprogramm initiiert wird, wird die Datei `/var/upgrade/status` entfernt (siehe „Statusdatei des zuverlässigen Aktualisierungsvorgangs“ auf Seite 97). Das heißt, dass der Status der vorherigen Aktualisierungssitzung (falls zutreffend) entfernt und durch den Status der neuen Aktualisierungssitzung ersetzt wird, sobald `rupgrade_tool -s` aufgerufen wird.

  Es können ein oder mehrere *--dst=src*-Bezeichner in einem Aufruf für `rupgrade_tool -s` enthalten sein. Jeder dieser Bezeichner definiert den Namen einer zu installierenden Aktualisierungsabbilddatei sowie den Installationsort im Flash-Speicher der Shelf-Verwaltungskarte.

*dst* definiert eines der folgenden Ziele für ein neu installiertes Aktualisierungsabbild :

- u – U-Boot-Abbild in der provisorischen U-Boot-Firmware-Abbildpartition aktualisieren (/dev/mtdchar3).
- u – Linux-Kernel-Abbild in der provisorischen Linux-Kernel-Abbildpartition aktualisieren (/dev/mtdchar2).
- u – RFS-Abbild in der provisorischen RFS-Abbildpartition aktualisieren (/dev/mtdchar4).

*src* gibt eine Aktualisierungsabbilddatei an, die in die durch *dst* angegebene provisorische Flash-Partition kopiert werden soll.

Die Abbildaktualisierung funktioniert wie folgt. Für jedes der angegebenen *src*-Abbilder wird das Abbild mithilfe des angegebenen Kopieprotokolls zu der Shelf-Verwaltungskarte kopiert. Wenn kein -d-Bezeichner angegeben ist, wird das Bild zuerst in das RAM-Dateisystem der Shelf-Verwaltungskarte kopiert (in das Verzeichnis /tmp) und dann zu Flash verschoben (d. h. es wird in die Zielpartition in Flash kopiert und dann aus dem RAM-Dateisystem entfernt). Wenn ein -d-Bezeichner beim Aufrufen von rupgrade_tool -s angegeben wird, erfolgt die vorübergehende Kopie in das Verzeichnis /tmp und das Abbild wird direkt in das Zielverzeichnis im Flash-Speicher kopiert. Die Verwendung dieses Bezeichners ist für ein Szenario vorgesehen, bei dem der Laufzeitspeicher auf der Shelf-Verwaltungskarte für ein vorübergehendes Kopieren in das RAM-Dateisystem nicht ausreichend ist.

Wenn kein -d-Bezeichner angegeben wird, ruft der zuverlässige Aktualisierungsvorgang ein spezielles Skript auf, dessen Hauptaufgabe darin besteht, Abbilder zu überprüfen, nachdem sie in das Verzeichnis /tmp kopiert wurden. Wenn ein -d-Bezeichner vorhanden ist, wird diese Überprüfung nicht durchgeführt.

Momentan führt das mit dem Shelf Manager bereitgestellte Skript /etc/upgrade/step4vshm keine spezifischen Schritte zur Abbildüberprüfung durch, ist jedoch verantwortlich für das Füllen der Flash-Partitionen, für die keine Abbilder im aktuellen Aufrufen von rupgrade_tool bereitgestellt sind (wie dies bei einem Szenario mit teilweiser Aktualisierung der Fall wäre). Diese Partitionen werden aus dem aktuellen persistenten Flash-Speicher in den provisorischen Flash-Speicher kopiert. Wenn die aktuelle teilweise Aktualisierung nur ein neues RFS-Abbild bereitstellt, kopiert das Skript die U-Boot- und Kernel-Partitionen aus dem persistenten Flash-Speicher in den provisorischen Flash-Speicher.

Sobald das erste Abbild im Zielverzeichnis installiert wurde, fährt das Dienstprogramm mit dem zweiten Abbild usw. fort (falls vorhanden), bis alle angegebenen Abbilddateien erfolgreich im Flash installiert wurden. Falls die erfolgreiche Installation eines Abbilds fehlschlägt, wird der Aktualisierungsvorgang sofort beendet (statt ein fehlerhaftes Abbild zu überspringen und mit dem nächsten fortzufahren).

Dies ermöglicht es dem Benutzer, die drei Teile der Firmware der Shelf-Verwaltungskarte (U-Boot-, Kernel- und RFS-Abbild) separat zu aktualisieren. Beachten Sie jedoch, dass die Teile, die nicht explizit aktualisiert werden, vom persistenten Flash-Speicher kopiert werden.

Es wird empfohlen, dass Sie eine der folgenden Aktualisierungsmethoden verwenden:

- Aktualisieren Sie explizit alle drei Partitionen.
- Wenn weniger als drei Partitionen explizit aktualisiert werden, lassen Sie den `-d`-Bezeichner weg. In diesem Fall stellt das oben erwähnte spezielle Skript sicher, dass jede Aktualisierung effektiv eine alle drei Partitionen umfassende vollständige Aktualisierung darstellt.

*Protokoll* gibt eines der folgenden Dateikopieprotokolle an, die verwendet werden, um jede der angegebenen *src*-Dateien in den Shelf Manager zu ziehen:

- `no` – Es wird kein Kopiervorgang durchgeführt. Dieses Protokoll setzt voraus, dass alle angegebenen *src*-Dateien in das Verzeichnis `/tmp` verschoben wurden, bevor der zuverlässige Aktualisierungsvorgang startet. Diese Protokollauswahl kann in Verbindung mit der Option `-d` verwendet werden.
- `cp:`*dir* – Einfache Kopie. Dieses Protokoll setzt voraus, dass alle angegebenen *src*-Dateien mithilfe des Befehls `cp` aus dem angegebenen Verzeichnis in das lokale Shelf Manager-Dateisystem kopiert werden sollen. Dieses Protokoll kann beispielsweise hilfreich sein, um Aktualisierungsabbilder aus einem NFS-Dateisystem oder aus einem JFFS2-Dateisystem zu installieren.
- ftp:*Server*:*dir*:*Benutzer*[:*pwd*] – Es wird von einem FTP-Remoteserver kopiert. Dieses Protokoll setzt voraus, dass alle angegebenen *src*-Dateien vom FTP-Hostserver, der vom Server als Hostname oder IP-Adresse angegeben ist, zu der Shelf-Verwaltungskarte kopiert werden sollen. Alle Abbilder müssen im Verzeichnis enthalten sein, das von *dir* auf dem FTP-Remoteserver angegeben wird. Die FTP-Verbindung wird mithilfe des vom Benutzerparameter angegebenen Kontos hergestellt, wobei das Passwort vom optionalen `pwd`-Parameter angegeben wird. Falls kein `pwd`-Parameter angegeben wird, fordert das Dienstprogramm zur Eingabe eines Passworts ein.

Falls das Kopieren eines Abbilds in den Shelf Manager fehlschlägt, beendet das Dienstprogramm den Aktualisierungsvorgang (statt ein fehlerhaftes Abbild zu überspringen und mit dem nächsten fortzufahren).

Für jede provisorische Flash-Partition, die mithilfe der Option `-s` aktualisiert wird, werden der zu aktualisierenden Partition Schreibberechtigungen erteilt, nachdem die Gültigkeit des Abbilds überprüft wurde und unmittelbar bevor das *src*-Abbild zum Flash-Speicher verschoben wird. Schreibberechtigungen werden sofort aus der Partition entfernt, nachdem das vollständige Abbild zum Flash-Speicher verschoben wurde. So wird zusammen mit der Tatsache, dass für U-Boot-, Linux-Kernel- und RFS-Abbilder nur Lesezugriff beim Starten des Shelf Manager besteht, sichergestellt, dass Anwendungen nicht aus Versehen wichtige Startabbilder löschen.

Nachdem alle angegebenen Abbilder in den jeweiligen Zielverzeichnissen im Flash-Speicher installiert wurden, ruft das Dienstprogramm ein *hook* (Eingriffspunkt)-Skript auf, das benutzerdefinierte Aktionen ermöglicht, die von einer Anwendung an dem Punkt benötigt werden, an dem die Aktualisierungsabbilder bereits im Flash-Speicher installiert wurden, aber der Aktualisierungsvorgang noch nicht die Hardwaremechanismen des zuverlässigen Aktualisierungsvorgangs initiiert hat, indem der Aktualisierungs-WDT (Watchdog Timer) des ShMM aktiviert wird.

Das „Hook"-Skript `/etc/upgrade/step4hshm` wird mit dem Shelf Manager bereitgestellt. Es führt die folgenden Aktionen durch:

- Beendet den Shelf Manager, indem ein Wechsel zum Backup-ShMM durchgeführt wird, ohne dass das Shelf neu gestartet wird. Der ATCA-Watchdog Timer wird gestoppt.
- Lädt die provisorischen Flash-Partitionen `/etc` und `/var` und löscht alle darin enthaltenen Dateien.
- Kopiert optional den aktuellen Inhalt vom Verzeichnis `/etc` in die provisorische Flash-Partition `/etc`.
- Kopiert optional die aktuellen nicht flüchtigen Shelf Manager-Informationen aus dem Verzeichnis `/var/nvdata` in das provisorische Dateisystem `/var` oder kopiert optional das gesamte `/var`-Verzeichnis in die provisorische Flash-Partition `/var`.
- Legt für die Startverzögerung temporär (bis zum nächsten Starten) `0` fest. Dies dient der Minimierung der Zeit bis zum nächsten Starten und verhindert das vorzeitige Ablaufen des Watchdog Timer der zuverlässigen Aktualisierung.

Das Skript wird als Subshell aufgerufen und ein einzelner Parameter angegeben, bei dem es sich entweder um die Zeichenkette handelt, die von *args* oder keinen *args* (leere Zeichenkette) angegeben wird. Die angegebenen Parameter definieren den Operationsmodus des Skripts, das Operationen wie das Kopieren von nicht flüchtigen Informationen aus den persistenten Flash-Partitionen in die provisorischen Flash-Partitionen bestimmt. Das Skript verwendet die folgenden *args*-Werte und führt die entsprechenden Aktionen durch:

- Kein Parameter bereitgestellt – Das Skript löscht die provisorischen Verzeichnisse `/etc` und `/var` und kopiert anschließend die nicht flüchtigen Shelf Manager-Informationen aus dem Verzeichnis `/var/nvdata` in die provisorische Partition `/var`. Dies ist der standardmäßige Operationsmodus. In diesem Fall bleiben die nicht flüchtigen Daten erhalten, aber die Shelf Manager-Konfigurationsdatei wird aus dem neuen RFS-Abbild herausgenommen.
- `erase` – das Skript löscht die provisorischen Verzeichnisse `/etc` und`/var`. Sie werden aus den RFS-Standardwerten beim nächsten Startvorgang wiederhergestellt. Die aktuellen nicht flüchtigen Shelf Manager-Daten und -Konfigurationen bleiben nicht erhalten.

- etc_copy – Das Skript löscht die provisorischen Verzeichnisse /etc und /var und kopiert anschließend den Inhalt von /etc und die nicht flüchtigen Informationen aus dem Verzeichnis /var/nvdata in die provisorischen Flash-Partitionen. In diesem Fall bleiben die nicht flüchtigen Daten und die Shelf Manager-Konfigurationsdatei erhalten.
- copy – Das Skript löscht die provisorischen Verzeichnisse /etc und /var und kopiert anschließend den vollständigen Inhalt der Verzeichnisse /etc und /var in die provisorische Partition. In diesem Fall werden nicht nur die Konfiguration, sondern auch die in /var/bin platzierten ausführbaren Dateien kopiert und überschreiben die ausführbaren Dateien mit demselben Namen aus dem RFS-Abbild. Dieser Operationsmodus ist hilfreich, wenn das Verzeichnis /var/bin einige spezielle ausführbare Dateien enthält (z. B. eine spezielle Version von Shelf Manager oder anderen Dienstprogrammen), die in der gesamten Aktualisierung erhalten bleiben müssen.

Das Skript gibt bei Erfolg 0 und bei einem Fehlschlagen einen Nichtnullwert zurück. Wenn ein Nichtnullwert zurückgegeben wird, wird der Aktualisierungsvorgang beendet.

Das Dienstprogramm startet den Aktualisierungs-WDT mit einer Zeitüberschreitungsperiode von 12,8 Sekunden. Diese Zeitüberschreitungsperiode wird für jede Software als ausreichend angesehen, die nach dem Zurücksetzen zu dem Punkt gestartet wird, an dem sie rupgrade_tool -c aufrufen kann (womit der Aktualisierungs-WDT geprüft wird, wenn er aktiv ist), ohne den Aktualisierungs-WDT in der Zwischenzeit prüfen zu müssen. Das Dienstprogramm führt ein Prüfen des Aktualisierungs-WDT durch, bevor das ShMM zurückgesetzt wird.

- -c [-v]

Fährt mit dem zuverlässigen Aktualisierungsvorgang fort, nachdem das ShMM vom provisorischen Flash-Speicher gestartet wurde. Das Aufrufen von rupgrade_tool -c wird über das Skript /etc/rc durchgeführt. Wie unten beschrieben, implizieren bestimmte von rupgrade_tool -c festgestellte Situationen einen Fehler beim Aktualisierungsvorgang und erfordern Korrekturmaßnahmen. Dazu zählen auch Fehler, die ein sanftes Zurücksetzen (Soft Reset) erfordern. Das heißt, dass auf ein Aufrufen von rupgrade_tool -c keine Rückgabe erfolgt, sondern unter Umständen das ShMM zurückgesetzt wird. Bei einem Zurücksetzen nimmt das ShMM den Status der Softwareinstallation im persistenten Flash-Speicher an.

Wenn der Aktualisierungs-WDT aktiv ist und während eines Schritts vor dem Aufrufen von rupgrade -c ausgelöst wurde, bedeutet dies, dass das ShMM bereits den Softwarestatus des persistenten Flash-Speichers angenommen hat. In diesem Szenario deaktiviert das Dienstprogramm den Aktualisierungs-WDT und kehrt zur Verwendung des persistenten Flash-Speichers zurück und beendet den Aktualisierungsvorgang.

Wenn der Aktualisierungs-WDT aktiv ist, aber nicht ausgelöst wurde, bedeutet dies, dass das ShMM (bis zu diesem Punkt) erfolgreich vom provisorischen Flash-Speicher gestartet wurde. Das Dienstprogramm prüft den Aktualisierungs-WDT und wird mit dem Ausgabecode 0 beendet, der darauf hinweist, dass momentan eine Aktualisierungssitzung ausgeführt wird.

Falls der Aktualisierungs-WDT nicht aktiv ist, aber der Inhalt der Datei /var/upgrade/status darauf hinweist, dass der Aktualisierungsvorgang noch ausgeführt wird, bedeutet dies, dass das ShMM aufgrund eines Systemneustarts bei einem der Schritte des Aktualisierungsvorgangs neu gestartet wurde. In diesem Szenario führt das Dienstprogramm dieselben Korrekturmaßnahmen durch wie in einer Situation, in der der Aktualisierungs-WDT aktiv ist und ausgelöst wurde.

Wenn der Aktualisierungs-WDT nicht aktiv ist und /var/upgrade/status entweder nicht vorhanden ist oder anzeigt, dass der Aktualisierungsvorgang abgeschlossen wurde (erfolgreich oder nicht erfolgreich), wird das Dienstprogramm mit dem Ausgabewert 1 beendet, der darauf hinweist, dass momentan kein Aktualisierungsvorgang ausgeführt wird.

- -f [--hook=args] [-v]

  Schließt den Aktualisierungsvorgang ab. Das Aufrufen von rupgrade_tool -f wird vom Shelf Manager aus durchgeführt, nachdem der Shelf Manager dessen Initialisierung erfolgreich abgeschlossen hat. Falls der Shelf Manager nicht automatisch gestartet wird, wird das Aufrufen am Ende des Skripts /etc/rc durchgeführt.

  Nach dem Aufrufen prüft rupgrade_tool -f den Aktualisierungs-WDT und fährt damit fort, den neuen persistenten Flash-Speicher einzurichten und den Aktualisierungs-WDT zu deaktivieren.

  Nach der Fertigstellung aktualisiert das Dienstprogramm /var/upgrade/status mit einem Datensatz, der ein erfolgreiches Abschließen des Aktualisierungsvorgangs anzeigt, und wird dann mit einem Wert von 0 beendet.

- -w [-f]

  Druckt den aktuellen Status des neuesten Aktualisierungsvorgangs. Im Wesentlichen gibt diese Option den Inhalt der Datei /var/upgrade/status in stdout aus.

  rupgrade_tool -w gibt einen Wert von 0 zurück, wenn der Aktualisierungsvorgang erfolgreich abgeschlossen wurde, und 1, wenn der Aktualisierungsvorgang nicht erfolgreich war, und einen Fehlercode, wenn /var/upgrade/status nicht vorhanden ist.

  Wenn der Bezeichner -f angegeben ist, entfernt rupgrade_tool -w die Datei /var/upgrade/status vor dem Beenden.

- -S [-v]

  Prüft den Aktualisierungs-WDT. rupgrade_tool -S dient als Shell-Ebenen-schnittstelle zur Verwendung durch neu installierte Software, die deren Gesamt-zustand überprüft.

  rupgrade_tool -S gibt einen Wert von 0 zurück.

- -u

  Macht eine erfolgreiche Aktualisierungssitzung rückgängig und stellt die Einstellungen des vorherigen persistenten Flash-Geräts wieder her.

  rupgrade_tool -u veranlasst das ShMM zu einem Neustart.

- -h

  Druckt Hilfe zu stdout.

## Szenarios für die Verwendung des Dienstprogramms für die zuverlässige Aktualisierung

Das Dienstprogramm für die zuverlässige Aktualisierung ist zur Verwendung in der folgenden Sequenz bestimmt, um eine Aktualisierung des ShMM durchzuführen:

1. Der Benutzer ruft rupgrade_tool -s auf, um den Aktualisierungsvorgang zu initiieren. Der Aufruf kann entweder lokal von der seriellen ShMM-Konsole oder remote über das Netzwerk mit telnet, rsh, ssh oder einem entsprechenden Dienstprogramm erfolgen.

2. Der Benutzer wartet auf rupgrade_tool -s, um das ShMM neu zu starten. Falls der Benutzer mit der seriellen Konsole lokal verbunden ist, ist der Neustartstatus über die Meldungen ersichtlich, die von der U-Boot-Firmware und Linux an die serielle Konsole ausgegeben werden. Bei einer Remoteverbindung mit der ShMM ist der Neustartstatus weniger klar ersichtlich. Bei einer Telnet-Verbindung tritt eine Zeitüberschreitung auf, wenn die Shelf-Verwaltungskarte neu gestartet wird. Der Benutzer kann entweder annehmen, dass der Aktualisierungsvorgang erfolgreich durchgeführt wurde, oder die Zeitspanne abwarten, die für das Abschließen der Aktualisierungssitzung erforderlich ist, und dann rupgrade_tool -w aufrufen (wieder remote mithilfe eines der oben erwähnten Remote-Shell-Tools), um den Status der Aktualisierungssitzung zu ermitteln. Die abzuwartende Zeitdauer hängt von der Größe der Aktualisierungsabbilder und vom Kopierprotokoll ab, das für das Ziehen der Abbilder in die Shelf-Verwaltungskarte verwendet werden, sowie von den Aktionen, die vom Abbildüberprüfungsskript durchgeführt werden.

3. Auf der Shelf-Verwaltungskarte führt das Startskript/etc/rc bedingungslos einen Aufruf für rupgrade_tool -c durch. Wenn der Aufruf einen Wert von 1 zurückgibt, der angibt, dass momentan keine Aktualisierung durchgeführt wird, oder einen Fehlercodewert, der angibt, dass die Aktualisierungssitzung

fehlgeschlagen ist, fahren die Startskripts mit der Startsequenz im normalen Modus fort. Wenn jedoch ein Wert von 0 zurückgegeben wird, der angibt, dass momentan eine Aktualisierungssitzung durchgeführt wird, fahren die Startskripts mit der Überprüfung des Gesamtzustands der neu installierten Software fort, rufen rupgrade_tool -S in der Mitte der Operation auf, um den Aktualisierungs-WDT zu prüfen, wenn die Überprüfung länger als die Zeitüberschreitungsperiode des WDT dauert, und starten schließlich den Shelf Manager, um die abschließende Überprüfung durchzuführen. Das Watchdog Timer-Intervall ist auf 12,8 Sekunden festgelegt. Daher darf die jeweilige Verarbeitungszeit im Skript /etc/rc zwischen dem Aufruf zu rupgrade_tool -c und dem Prüfen des WDT sowie zwischen dem Prüfen des WDT und dem Starten des Shelf Managers die Dauer von 12,8 Sekunden nicht überschreiten.

4. Während der Initialisierung prüft der Shelf Manager den Aktualisierungs-WDT erneut, bevor er versucht, eine Netzwerkverbindung mit dem gleichgeordneten Shelf Manager herzustellen. Das Herstellen einer Netzwerkverbindung kann bis zu sechs Sekunden dauern. Danach und nach dem erfolgreichen Abschließen der Initialisierung (was die Gültigkeit der neuen Konfiguration bestätigt) ruft der Shelf Manager rupgrade_tool -f auf, wodurch der Aktualisierungsvorgang fertig gestellt wird.

5. Der Benutzer ruft optional rupgrade_tool -w auf, um den Status der Aktualisierungssitzung zu ermitteln. Wie oben beschrieben kann diese Option besonders hilfreich für eine Remote-Aktualisierungssitzung sein, bei der der Aktualisierungsfortschritt nicht direkt über Meldungen verfolgt werden kann, die an die serielle Konsole ausgegeben werden, wie dies bei einer lokalen Aktualisierung der Fall ist.

Nachdem die zuverlässige Aktualisierung abgeschlossen ist, kann der Benutzer die ursprünglichen Abbilder wiederherstellen, falls die neuen Abbilder aus bestimmten Gründen nicht akzeptabel sind. Dazu ruft der Benutzer rupgrade_tool -u auf.

Falls erforderlich, kann der obige Ablauf problemlos automatisiert werden, indem ein einfaches Skript entwickelt wird, das auf einem Remote-Netzwerkhost ausgeführt werden soll. Alternativ dazu kann ein Benutzer die zuverlässige Aktualisierung entweder lokal über die serielle Konsole oder remote über das Netzwerk manuell ausführen.

## Beispiele für die zuverlässige Aktualisierung

### *Beispiel 1:*

Dieses Beispiel zeigt eine zuverlässige Aktualisierung aller drei Komponenten (U-Boot-, Kernel- und RFS-Abbild), bei der die nicht flüchtigen Verzeichnisse /etc und /var/nvdata in den provisorischen Flash-Speicher kopiert werden. Alle

Abbilder wurden aus der lokalen Datei /tmp entnommen (was bedeutet, dass sie auf unspezifische Weise bereits dorthin kopiert wurden). Das U-Boot-Abbild wurde aus /tmp/u-boot.bin, das Kernel-Abbild aus /tmp/sentry.kernel und das RFS-Abbild wurde aus /tmp/sentry.rfs entnommen. Der Aktualisierungsvorgang wird über die serielle Konsole gestartet. Im Konsolenprotokoll werden Kommentare eingestreut, um zusätzliche Hintergrundinformationen zu den Schritten des Aktualisierungsvorgangs bereitzustellen.

Zuerst wird rupgrade_tool über die Eingabeaufforderung gestartet. Die Parameter zeigen, dass alle drei Flash-Abbilder aktualisiert werden sollen, wobei die nicht flüchtigen Daten und die Konfigurationsdatei des Shelf Managers ebenfalls erhalten bleiben sollen.

```
# rupgrade_tool -s --k=sentry.kernel --r=sentry.rfs
--u=u-boot.bin --hook=etc_copy -v
rupgrade_tool: PLB is 5
rupgrade_tool: EEPROM page saved
rupgrade_tool: persistent flash is 0
rupgrade_tool: provisional flash is 1
rupgrade_tool: copying image(s)
```

Das Aktualisierungsdienstprogramm versucht, ein Überprüfungsskript aufzurufen, um die in /tmp momentan bereitgestellten Abbilder zu prüfen. Falls einer der angegebenen Dateibezeichner in /tmp nicht gefunden wird, wird das Dienstprogramm beendet und eine Meldung ähnlich der folgenden wird erstellt.

```
rupgrade_tool: cannot open /tmp/u-boot.bin for reading.
rupgrade_tool: failed to copy images to flash
```

Das Dienstprogramm fährt damit fort, die Abbilder in die angegebenen Zielverzeichnisse im provisorischen Flash-Speicher zu kopieren.

```
rupgrade_tool: invoking scripts (step4v*) [--u=u-boot.bin --k=
sentry.kernel --r=sentry.rfs --hook=etc_copy]
rupgrade_tool: copying u-boot.bin from /tmp to /dev/mtdchar8 using
'cp' protocol
rupgrade_tool: copying sentry.kernel from /tmp to /dev/mtdchar7
using 'cp' protocol
rupgrade_tool: copying sentry.rfs from /tmp to /dev/mtdchar9 using
'cp' protocol
rupgrade_tool: invoking scripts (step4h*) [etc_copy]
```

An diesem Punkt wird das Eingriffspunkt (Hook)-Skript `step4hshm` aufgerufen. Es beendet den Shelf Manager und kopiert nicht flüchtige Informationen in den provisorischen Flash-Speicher.

```
/etc/upgrade/step4hshm: Stopping Shelf Manager...
/etc/upgrade/step4hshm: Erasing /var and /etc, copying
/var/nvdata...
/etc/upgrade/step4hshm: Operation: copy /etc and /var/nvdata.
/etc/upgrade/step4hshm: Copying completed.
rupgrade_tool: image(s) copy OK
rupgrade_tool: watchdog started
rupgrade_tool: selected provisional flash
rupgrade_tool: reboot
Restarting system.
```

Hier setzt der zuverlässige Aktualisierungsvorgang das ShMM zurück. Dies veranlasst U-Boot, vom provisorischen Flash-Speicher zu starten.

```
* Resetting Integrated Peripherals

U-Boot 1.1.2 (May 12 2005 - 21:27:13)

CPU: Au1550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 8000044
DRAM:  128 MB
Flash: 64 MB
In:    serial
Out:   serial
Err:   serial
Net:   Au1X00 ETHERNET
Hit any key to stop autoboot:  0
## Booting image at bfb00000 ...
   Image Name:   MIPS Linux-2.4.26
   Created:      2005-06-24  13:29:50 UTC
   Image Type:   MIPS Linux Kernel Image (gzip compressed)
   Data Size:    844843 Bytes = 825 kB
   Load Address: 80100000
   Entry Point:  802bc040
   Verifying Checksum ... OK
   Uncompressing Kernel Image ... OK
## Loading Ramdisk Image at bfc40000 ...
   Image Name:   sentry RFS Ramdisk Image
   Created:      2005-04-22   9:10:41 UTC
   Image Type:   MIPS Linux RAMDisk Image (gzip compressed)
```

```
   Data Size:    2465924 Bytes =  2.4 MB
   Load Address: 00000000
   Entry Point:  00000000
   Verifying Checksum ... OK

Starting kernel ...

init started:  BusyBox v0.60.5 (2005.06.15-14:45+0000) multi-call
binary
/etc/rc: Mounted /proc
/etc/rc: Mounting filesystems...
/etc/rc: Mounted /dev/pts
/etc/rc: Mounted /dev/mtdblock0 to /var
/etc/rc: Mounted /dev/mtdblock10 to /var/upgrade
```

An diesem Punkt ruft das Skript `rc` `rupgrade_tool -c` auf, um zu prüfen, ob momentan eine zuverlässige Aktualisierung durchgeführt wird. Das Tool gibt `0` zurück, um zu bestätigen, dass eine Aktualisierung momentan durchgeführt wird. Bei Angabe dieses Ergebnisses fährt das Skript `rc` mit dem Startvorgang fort.

```
/etc/rc: Checking the reliable upgrade watchdog timer
/etc/rc: Mounted ram disk to /var/log
/etc/rc: Started syslogd and klogd
/etc/rc: Mounted ram disk to /var/tmp
/etc/rc: Setting hostname shmm+193
```

Da momentan eine zuverlässige Aktualisierung durchgeführt wird, wird der Watchdog Timer erneut im Skript `rc` geprüft.

```
/etc/rc: Strobing the reliable upgrade watchdog timer
/etc/rc: Mounted /dev/mtdblock1 to /etc
/etc/rc: Calling /etc/rc.carrier3
Board Hardware Address: 0xFE
/etc/netconfig: /etc/hosts has valid  192.168.1.193 entry
/etc/netconfig: Updating /etc/profile.sentry with IP settings
/etc/netconfig: ifconfig eth0 192.168.1.193
/etc/netconfig: ifconfig eth1 192.168.0.193
/etc/netconfig: route add default gw 192.168.1.253
/etc/netconfig: Starting inetd...
/etc/rc.carrier3: Starting up IPMBs ...
/etc/rc.carrier3: Updating /etc/profile.sentry with specific
settings
/etc/rc.carrier3: Starting snmpd...
/etc/rc.carrier3: Starting httpd...
/etc/rc.carrier3: Starting Shelf Manager ...
```

```
<I> 02:48:08.463  [171] IPM Sentry Shelf Manager ver. 2.0.0. Built
on Jun 27 2005 14:48:57
<*> 02:48:08.469   [171] Limits: code=(400000:506f0), end_data=
10062000, start_stack=7fff7e30, esp=7fff78a0, eip=2ab0d2e4
<*> 02:48:08.469   [171] Stack limits: curr=1ff000, max=7fffffff
<*> 02:48:08.470   [171] Data limits: curr=7fffffff, max=7fffffff
<*> 02:48:08.900   [171] *** Lock log print buffer at 1003b7f0 ***
<*> 02:48:08.900   [171] *** Pthread lock log print buffer at
1003f820 ***
```

Der Shelf Manager startet die zuverlässige Aktualisierung und stellt sie fertig, in dem er `rupgrade_tool -f` ausführt.

```
eth0: link up
eth1: link up
eth1: going to full duplex

shmm+193 login:root

BusyBox v0.60.5 (2005.05.12-22:46+0000) Built-in shell (msh)
```

Der Benutzer prüft dann den Status der zuverlässigen Aktualisierung, indem er den Befehl `rupgrade_tool -w` ausgibt.

```
# rupgrade_tool -w
Recent upgrade status:
1:PLB is 5
1:EEPROM page saved
2:persistent flash is 1
3:provisional flash is 0
4:copying image(s)
4:invoking scripts (step4v*) [--u=u-boot.bin --k=sentry.kernel --
r=sentry.rfs  --hook=etc_copy]
4:copying u-boot.bin from /tmp to /dev/mtdchar8 using 'cp'
protocol
4:copying sentry.kernel from /tmp to /dev/mtdchar7 using 'cp'
protocol
4:copying sentry.rfs from /tmp to /dev/mtdchar9 using 'cp'
protocol
4:invoking scripts (step4h*) [etc_copy]
4:image(s) copy OK
5:watchdog started
6:selected provisional flash
7:reboot
9:WDT not fired, upgrade in progress.
11:provisional flash 0, updating EEPROM
```

```
12:EEPROM updated
13:upgrade WDT disabled
13:invoking scripts (step13h*) []
14:upgrade completed successfully
#
```

*Beispiel 2:*

Dieses Beispiel zeigt eine zuverlässige Aktualisierung lediglich des RFS-Abbilds, bei der die nicht flüchtigen Verzeichnisse `/etc` und `/var/nvdata` in den provisorischen Flash-Speicher kopiert werden. Das RFS-Abbild wird aus einem FTP-Server mit der IP-Adresse `192.168.1.253` entnommen. Der Pfad zum RFS-Abbild auf dem FTP-Server lautet `/tftpboot/ru-mips/sentry.mips.rfs`. Der Aktualisierungsvorgang wird über eine Telnet-Sitzung gestartet.

---

**Hinweis –** Da nur das RFS-Abbild explizit aktualisiert wird, werden die U-Boot- und Kernel-Abbilder automatisch aus der persistenten Flash-Partition in die provisorische Flash-Partition kopiert.

---

Das lokale System muss über das Netzwerk Zugriff auf den FTP-Server haben (d h. die Netzwerkadapter müssen ausgeführt werden und vom ShMM zum FTP-Server muss eine Route bestehen). In dem Beispiel wird ShMM mit der Netzwerkadresse `192.168.1.174` konfiguriert (im selben Netzwerk wie der FTP-Server):

```
# telnet 192.168.1.174
Trying 192.168.1.174...
Connected to 192.168.1.174.
Escape character is '^]'.

BusyBox on shmm+174 login: root


BusyBox v0.60.5 (2005.05.07-17:27+0000) Built-in shell (msh)
```

Die Parameter zu `rupgrade_tool -s` zeigen an, dass nur das RFS aktualisiert wird, dass das Kopierprotokoll FTP ist, dass auf eine spezifische IP-Adresse und Datei zugegriffen wird, dass der Benutzer `admin` lautet und dass kein Passwort bereitgestellt ist.

```
# rupgrade_tool -s --r=sentry.mips.rfs
--proto=ftp:192.168.1.253:/tftpboot/ru-mips:admin --hook=etc_copy
-v
rupgrade_tool: PLB is 5
rupgrade_tool: EEPROM page saved
```

```
rupgrade_tool: persistent flash is 1
rupgrade_tool: provisional flash is 0
rupgrade_tool: copying image(s)
rupgrade_tool: copying sentry.rfs from
192.168.1.253:/tftpboot/ru-mips:admin to /tmp using 'ftp' protocol
220 hydra FTP server (Version wu-2.4.2-academ[BETA-17](1) Tue Jun
9 10:43:14 EDT 1998) ready.
USER admin
```

Der Benutzer wird hier nach einem Passwort für die FTP-Site gefragt. Das Passwort wird manuell eingegeben.

```
331 Password required for admin.
Password:xxxx
PASS *****
230 User admin logged in.
TYPE I
200 Type set to I.
PASV
227 Entering Passive Mode (192,168,1,253,9,20)
RETR /tftpboot/ru-mips/sentry.mips.rfs
150 Opening BINARY mode data connection for
/tftpboot/ru-mips/sentry.mips.rfs (2465988 bytes).
226 Transfer complete.
QUIT
221 Goodbye.
```

Im nächsten Schritt wird das spezielle Skript `step4vshm` aufgerufen, dass die U-Boot- und Kernel-Abbilder aus dem persistenten Flash-Speicher in den provisorischen Flash-Speicher kopiert. Anschließend fährt das Aktualisierungsdienstprogramm mit dem Kopieren des RFS-Abbilds in die zugewiesene Position im provisorischen Flash-Speicher fort.

```
rupgrade_tool: invoking scripts (step4v*) [--r=sentry.mips.rfs --
proto=ftp:192.168.1.253:/tftpboot/ru-mips:admin --hook=etc_copy]
/etc/upgrade/step4vshm: Erasing /dev/mtdchar7...Done
etc/upgrade/step4vshm: Copying Kernel from /dev/mtdchar2 to
/dev/mtdchar7...Done
/etc/upgrade/step4vshm: Erasing /dev/mtdchar8...Done
/etc/upgrade/step4vshm: Copying U-Boot from /dev/mtdchar3 to
/dev/mtdchar8...Done
rupgrade_tool: copying sentry.mips.rfs from /tmp to /dev/mtdchar9
using ’cp’ protocol
```

Das Eingriffspunkt-Skript `step4hshm` wird aufgerufen, das den Shelf Manager beendet und die nicht flüchtigen Daten beibehält. Das Dienstprogramm startet den Aktualisierungs-WDT und führt einen Neustart durch.

```
rupgrade_tool: invoking scripts (step4h*) [etc_copy]
/etc/upgrade/step4hshm: Stopping Shelf Manager...
/etc/upgrade/step4hshm: Erasing /var and /etc, copying
/var/nvdata..
/etc/upgrade/step4hshm: Operation: copy /etc and /var/nvdata.
/etc/upgrade/step4hshm: Copying completed.
rupgrade_tool: image(s) copy OK
rupgrade_tool: watchdog started
rupgrade_tool: selected provisional flash
rupgrade_tool: reboot
Restarting system.
Connection closed by foreign host.
```

An diesem Punkt wird die Telnet-Sitzung nach einem bestimmten Inaktivitätszeitraum geschlossen. Nach einigen Sekunden ist es möglich, wieder eine Verbindung zum Ziel herzustellen und den Status der zuverlässigen Aktualisierung zu prüfen, indem `rupgrade_tool -w` aufgerufen wird.

```
# telnet 192.168.1.174
Trying 192.168.1.174...
Connected to 192.168.1.174.
Escape character is '^]'.

BusyBox on shmm+174 login: root

BusyBox v0.60.5 (2005.05.07-17:27+0000) Built-in shell (msh)
#
# rupgrade_tool -w
Recent upgrade status:
1:PLB is 5
1:EEPROM page saved
2:persistent flash is 1
3:provisional flash is 0
4:copying image(s)
4:copying sentry.mips.rfs from 192.168.1.253:/tftpboot/ru-
mips:admin to /tmp using 'ftp' protocol
4:invoking scripts (step4v*) [--r=sentry. rfs --hook=etc_copy]
4:copying sentry.mips.rfs from /tmp to /dev/mtdchar9 using 'cp'
protocol
4:invoking scripts (step4h*) [etc_copy]
4:image(s) copy OK
5:watchdog started
```

```
6:selected provisional flash
7:reboot
9:WDT not fired, upgrade in progress.
11:provisional flash 0, updating EEPROM
12:EEPROM updated
13:upgrade WDT disabled
13:invoking scripts (step13h*) []
14:upgrade completed successfully
#
```

*Beispiel 3:*

Dieses Beispiel zeigt eine nicht erfolgreiche zuverlässige Aktualisierung. Die Stromversorgung wird nach dem Starten vom provisorischen Flash-Speicher aber vor dem Abschließen der zuverlässigen Aktualisierung ausgeschaltet. Nach dem erneuten Einschalten tritt die Rollback-Funktion (Rücksetzung) für den persistenten Flash-Speicher auf. Die zuverlässige Aktualisierung wird über die serielle Konsole initiiert. Es wird davon ausgegangen, dass sich alle drei Abbilder bereits in `/tmp` befinden.

```
# rupgrade_tool -s --k=sentry.kernel --r=sentry.rfs --u=u-boot.bin
--hook=etc_copy -v
rupgrade_tool: PLB is 5
rupgrade_tool: EEPROM page saved
rupgrade_tool: persistent flash is 0
rupgrade_tool: provisional flash is 1
rupgrade_tool: copying image(s)
rupgrade_tool: invoking scripts (step4v*) [--u=u-boot.bin --k=
sentry.kernel --r=sentry.rfs --hook=etc_copy]
rupgrade_tool: copying u-boot.bin from /tmp to /dev/mtdchar8 using
'cp' protocol
rupgrade_tool: copying sentry.kernel from /tmp to /dev/mtdchar7
using 'cp' protocol
rupgrade_tool: copying sentry.rfs from /tmp to /dev/mtdchar9 using
'cp' protocol
rupgrade_tool: invoking scripts (step4h*) [etc_copy]
Stopping Shelf Manager...

IPM Sentry Shelf Manager Command Line Interpreter

Terminating the

Erasing /var and /etc, copying /var/nvdata...
Operation: copy /etc and /var/nvdata.
Copying completed.
```

```
rupgrade_tool: image(s) copy OK
rupgrade_tool: watchdog started
rupgrade_tool: selected provisional flash
rupgrade_tool: reboot
Restarting system.
```

Der zuverlässige Aktualisierungsvorgang setzt das ShMM hier zurück und startet U-Boot vom provisorischen Flash-Speicher.

```
** Resetting Integrated Peripherals


U-Boot 1.1.2 (Nov 11 2005 - 15:16:25)

CPU: Au1550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 8000044
DRAM:  128 MB
Flash: 64 MB
In:    serial
Out:   serial
Err:   serial
Net:   Au1X00 ETHERNET
Hit any key to stop autoboot:  0
## Booting image at bfb00000 ...
   Image Name:   MIPS Linux-2.4.26
   Created:      2005-04-11  10:35:08 UTC
   Image Type:   MIPS Linux Kernel Image (gzip compressed)
   Data Size:    843129 Bytes = 823.4 kB
   Load Address: 80100000
   Entry Point:  802bc040
   Verifying Checksum ... OK
   Uncompressing Kernel Image ... OK
## Loading Ramdisk Image at bfc40000 ...
   Image Name:   sentry RFS Ramdisk Image
   Created:      2005-04-22   9:10:41 UTC
   Image Type:   MIPS Linux RAMDisk Image (gzip compressed)
   Data Size:    2400736 Bytes =  2.3 MB
   Load Address: 00000000
   Entry Point:  00000000
   Verifying Checksum ... OK
```

Hier wird die Stromversorgung ausgeschaltet. Nach einiger Zeit wird wieder eingeschaltet. Die Zuordnung des provisorischen Flash-Speichers ging aufgrund der Stromunterbrechung verloren. Daher nimmt das System wieder die Einstellungen des persistenten Flash-Speichers an.

```
U-Boot 1.1.2 (Nov 11 2005 - 15:16:25)

CPU: Au1550 324 MHz, id: 0x02, rev: 0x00
Board: ShMM-500
S/N: 8000048
DRAM:  128 MB
Flash: 64 MB
In:    serial
Out:   serial
Err:   serial
Net:   Au1X00 ETHERNET
Hit any key to stop autoboot:  0
## Booting image at bfb00000 ...
   Image Name:   MIPS Linux-2.4.26
   Created:      2005-04-11  10:35:08 UTC
   Image Type:   MIPS Linux Kernel Image (gzip compressed)
   Data Size:    843129 Bytes = 823.4 kB
   Load Address: 80100000
   Entry Point:  802bc040
   Verifying Checksum ... OK
   Uncompressing Kernel Image ... OK
## Loading Ramdisk Image at bfc40000 ...
   Image Name:   sentry RFS Ramdisk Image
   Created:      2005-04-11  18:27:17 UTC
   Image Type:   MIPS Linux RAMDisk Image (gzip compressed)
   Data Size:    2372311 Bytes =  2.3 MB
   Load Address: 00000000
   Entry Point:  00000000
   Verifying Checksum ... OK

Starting kernel ...

init started:  BusyBox v0.60.5 (2005.02.07-16:45+0000) multi-call
binary
hub.c: new USB device AU1550-1, assigned address 2
usb0: ? speed config #1: Ethernet Gadget
usb1: register usbnet usb-AU1550-1, Linux Device
serial#=8000048: not found
/etc/rc: Mounted /proc
/etc/rc: Mounting filesystems...
```

```
/etc/rc: Mounted /dev/pts
/etc/rc: Mounted /dev/mtdblock0 to /var
/etc/rc: Mounted /dev/mtdblock10 to /var/upgrade
```

Als nächster Schritt im `rc`-Skript wird `rupgrade_tool -c` aufgerufen, um zu prüfen, ob momentan eine zuverlässige Aktualisierung durchgeführt wird. Die Prüfung ergibt, dass der Versuch, eine zuverlässige Aktualierung durchzuführen, fehlgeschlagen ist. Die Meldung `restoring ADM1060 EEPROM to RAM` bezieht sich auf das ShMM-Systemüberwachungsgerät (ein ADM1060), das den ShMM-Startvorgang überwacht und einige Hardware-Aspekte der Unterstützung für die zuverlässigen Aktualisierung implementiert. Diese Meldung gibt an, dass wichtige Variablen, die den Startvorgang betreffen, in deren Zustand vor der versuchten zuverlässigen Aktualisierung wiederhergestellt werden.

```
/etc/rc: Checking the reliable upgrade watchdog timer
rupgrade_tool: Watchdog not active.
rupgrade_tool: restoring ADM1060 EEPROM to RAM
rupgrade_tool: upgrade failed
/etc/rc: Rupgrade -c Ret: 255
/etc/rc: Mounted ram disk to /var/log
/etc/rc: Started syslogd and klogd
/etc/rc: Mounted ram disk to /var/tmp
/etc/rc: Setting hostname shmm+173
/etc/rc: Mounted /dev/mtdblock1 to /etc
/etc/rc: Calling /etc/rc.carrier3
Board Hardware Address: 0xFE
/etc/netconfig: /etc/hosts has valid  192.168.1.173 entry
/etc/netconfig: Updating /etc/profile.sentry with IP settings
/etc/netconfig: Starting inetd...
/etc/rc.carrier3: Starting up IPMBs ...
/etc/rc.carrier3: Updating /etc/profile.sentry with specific
settings
/etc/rc.carrier3: RC2 daemons not started by request
```

# Programmieren des CPLD

Das ShMM beinhaltet ein CPLD-Gerät. Dieses CPLD-Gerät ist zuständig für die Steuerung verschiedener Schlüsselaspekte der ShMM-Operation, wie z. B. die Hardware-Redundanzschnittstelle.

## ▼ So programmieren Sie das CPLD-Abbild für das ShMM um

1. **Laden Sie die folgenden Dateien in das Verzeichnis `/var/bin` Ihrer Shelf-Verwaltungskarte hoch:**
   - `cpld_tool`
   - `shmm-cpld-erase.xsvf`
   - `shmm-cpld-1x.xsvf`
   - `shmm-cpld.xsvf`

2. **Löschen Sie das CPLD wie folgt:**

```
# cpld_tool -x shmm-cpld-erase.xsvf
```

3. **Programmieren Sie das CPLD mit einem entsprechenden Abbild:**

```
# cpld_tool -x shmm-cpld.xsvf
```

4. **Prüfen Sie die Gültigkeit des CPLD-Abbilds:**

   Die Benutzer-ID sollte gleich 0x33623030 sein:

```
# cpld_tool -u
 0x33623030
```

# Herstellen einer Verbindung zu der Node Board-Konsole

Mit dem Shelf Manager können Sie eine Verbindung zu Node Boards und offenen Konsolensitzungen über die aktive Shelf-Verwaltungskarte (ShMM) herstellen. Zuerst melden Sie sich beim aktiven ShMM entweder über den seriellen Port oder den Ethernet-Port an.

---

**Hinweis –** Die primäre (oder obere) Shelf-Verwaltungskarte muss die **aktive** Shelf-Verwaltungskarte zur Verwendung der Konsolenfunktion sein. Eine Switch-Karte muss ebenfalls in Steckplatz 7 der Mittelplatine des Netra CT 900 Server installiert sein.

---

Sobald eine Konsolensitzung mit einem Node Board eingerichtet wurde, können Sie Systemadministrationsbefehle wie `passwd` ausführen, Status- und Fehlermeldungen lesen oder das Board in diesem Steckplatz anhalten.

---

**Hinweis –** Wenn eine Konsole oder ein serielles Kabel an den seriellen Port des Node Boards angeschlossen wird, erfolgt die Konsolenausgabe zu der mit dem Kabel verbundenen Konsole statt zu der Konsolensitzung auf dem ShMM. Dies ist selbst dann der Fall, wenn die Konsolensitzung des ShMM aktiv war, als das Kabel angeschlossen wurde.

---

# Einrichten von Konsolensitzungen zwischen Shelf Manager und Node Boards

Nachdem Sie Ihr System für die Konsolenverwendung konfiguriert haben, können Sie sich beim aktiven ShMM anmelden und eine Konsole für einen Steckplatz öffnen. Das Netra CT-System lässt eine Konsolensitzung je Node Board-Steckplatz zu.

TABELLE 3-4 enthält die Befehle für die Shelf Manager-CLI-Konsole, die in der aktuellen Anmeldungssitzung auf der Shelf-Verwaltungskarte ausgeführt werden können.

**TABELLE 3-4** Befehle für die Shelf Manager-CLI-Konsole

| Befehl | Beschreibung |
|---|---|
| `clia console` *Steckplatz_Nr* | In den Konsolenmodus wechseln und das Node Board mit *Steckplatz_Nr* verbinden, wobei *Steckplatz_Nr* die Steckplatz-Nr. darstellt, in der sich das Node Board befindet |
| `~q oder ~.` | Verbindung der aktuellen Konsolensitzung trennen |

## ▼ So starten Sie eine Konsolensitzung über den Shelf Manager

1. **Melden Sie beim primären (oder oberen) ShMM an.**

   Sie können sich beim primären (oder oberen) ShMM über ein Terminal anmelden, das entweder über die serielle Port- oder die Ethernet-Portverbindung verbunden ist.

2. **Prüfen Sie, ob das primäre ShMM das aktive ShMM ist.**

   Nachdem Sie sich angemeldet haben, verwenden Sie den Befehl `clia shmstatus`, um zu prüfen, ob Sie beim *aktiven* ShMM angemeldet sind, bevor Sie fortfahren. Wenn Sie beim *Standby*-ShMM angemeldet sind, verwenden Sie den Befehl `clia switchover`, um das ShMM zu `Aktiv` zu wechseln (weitere Informationen finden Sie unter `shmstatus` und `switchover` in „Shelf Manager-CLI-Befehle“ auf Seite 125 .

3. **Öffnen Sie eine Konsolensitzung zu einem Node Board.**

```
# clia console Steckplatz_Nr
```

wobei *Steckplatz_Nr* für 1 - 6 und 9 - 14 steht. Um z. B. eine Konsole zum Board in Steckplatz 4 zu öffnen, geben Sie Folgendes ein:

```
# clia console 4
```

Sie haben jetzt Zugriff auf das Node Board in Steckplatz 4. Je nach Status des Boards in diesem Steckplatz und abhängig davon, ob sich der vorherige Benutzer bei der Shell abgemeldet hat, wird eine der folgenden Eingabeaufforderungen angezeigt:

- `console login%` (Solaris-Ebene)
- `#` (Solaris-Ebene; vorheriger Benutzer hat sich als Superuser angemeldet und hat sich nicht abgemeldet, bevor er die Verbindung zur Konsole beendet hat.)
- `ok` (OpenBoot PROM-Ebene)
- `#` (Monta Vista Linux)

## ▼ So beenden Sie die Konsolensitzung

1. **(Optional) Melden Sie sich bei der OS-Shell ab.**

2. **Trennen Sie bei der Eingabeaufforderung die Verbindung zur Konsole, indem Sie die Escape-Sequenz `~q` oder `~.` (Tilde und Punkt) eingeben.**

```
Eingabeaufforderung ~q
```

Wenn Sie die Verbindung zur Konsole trennen, werden Sie nicht automatisch vom Remote-Host abgemeldet. Wenn Sie sich nicht beim Remote-Host abmelden, sieht der nächste Konsolenbenutzer, der eine Verbindung zu diesem Board herstellt, die Shell-Eingabeaufforderung Ihrer vorherigen Sitzung.

---

**Hinweis –** Melden Sie sich immer bei der Konsolensitzung ab, wenn Sie fertig sind.

---

# Manuelles sanftes Herunterfahren der Node Boards

Für diese Version wurde eine Funktion für das manuelle sanfte Herunterfahren bereitgestellt, um das versehentliche Entfernen von Node Boards oder FRUs zu verhindern. Sanftes Herunterfahren bedeutet, dass alle auf der Nutzlast des Node Boards ausgeführten Anwendungen und das Betriebssystem sowie die Nutzlast selbst auf eine nicht abrupte Weise in den Ruhezustand gebracht (heruntergefahren) werden.

Vor dem Austauschen während des laufenden Betriebs bzw. dem Entfernen eines Node Boards von Netra sollten die auf dem Node Board ausgeführten Anwendungen und das Betriebssystem sanft heruntergefahren werden. Mithilfe der Netzkonsolenfunktion von Shelf Manager kann der Systemadministrator eine Konsolensitzung auf dem Node Board starten und ein sanftes Herunterfahren entsprechend Verfahren des Standorts zum Herunterfahren durchführen.

Nachdem das Herunterfahren abgeschlossen ist, aktiviert der Systemadministrator die Deaktivierung des FRU (oder Node Boards) und öffnet die Hot-Swap-Arretierung für das Board. Nachdem das Node Board ausgetauscht oder erneut installiert wurde, deaktiviert der Systemadministrator die Deaktivierung des FRU (oder Node Boards), um ein unsanftes Herunterfahren zu verhindern, während die Hot-Swap-Arretierung geöffnet ist.

Die für diesen Vorgang erforderlichen Schritte werden in den folgenden Verfahren ausführlich beschrieben.

## ▼ So fahren Sie ein Node Board herunter

Für dieses Verfahren ist es erforderlich, dass die Hot-Swap-Arretierung für das Node Board geschlossen ist.

---

**Hinweis –** Wenn die Hot-Swap-Arretierung geöffnet ist, blinkt die blaue Hot-Swap-LED fortlaufend und das Board wird nicht bereit für einen Hot-Swap-Vorgang (durch eine ununterbrochen leuchtende blaue LED angezeigt): Um diesen Zustand zu korrigieren, schließen Sie die Hot-Swap-Arretierung, bevor Sie mit diesem Verfahren beginnen.

---

1. **Melden Sie sich beim aktiven ShMM an.**
2. **Starten Sie eine Konsolensitzung mit einem Node Board.**

   Siehe „So starten Sie eine Konsolensitzung über den Shelf Manager“ auf Seite 120.

3. **Fahren Sie die Anwendung und das Betriebssystem auf dem Node Board herunter.**

   Befolgen Sie die Verfahren Ihres Standorts für das sanfte Herunterfahren. Nachdem das Herunterfahren abgeschlossen ist, schließen Sie die Konsolensitzung (siehe So beenden Sie die Konsolensitzung).

4. **Aktivieren Sie die kontrollierte Deaktivierung von Shelf Manager für das Node Board, indem Sie den folgenden CLI-Befehl von Shelf Manager verwenden.**

```
# clia shelf deactivate Hardware-Adr FRU-ID 0
```

   Beispiel:

```
# clia shelf deactivate 0x41 0xfe 0
IPM Sentry Shelf Manager Command Line Interpreter
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info, address: 0x41, FRU ID # 254
    Cached information updated
    Wrote Information to the Shelf FRU
#
```

5. **Prüfen Sie, dass die vom Shelf Manager kontrollierte Deaktivierung für das Node Board aktiviert ist, indem Sie den folgenden CLI-Befehl von Shelf Manager verwenden.**

```
# clia shelf pm
```

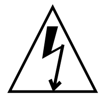
   Suchen Sie die Meldung `Shelf Manager Controlled Deactivation: Enabled`.

   Beispiel:

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
        Version = 1
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 16
    Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Shelf Manager Controlled Deactivation: Enabled
        Delay Before Next Power On: 0.0 seconds
```

6. **Öffnen Sie die Hot-Swap-Arretierung für das Node Board.**

7. **Wenn die blaue Hot-Swap-LED für das Node Board ununterbrochen leuchtet, entfernen oder ersetzen Sie das Node Board.**



**Achtung –** Tragen Sie beim Umgang mit Modulen immer ein geerdetes Antistatik-Armband.

8. **Wenn das Node Board ausgetauscht oder erneut installiert wird, deaktivieren Sie die vom Shelf Manager kontrollierte Deaktivierung für das Node Board, indem Sie den folgenden CLI-Befehl von Shelf Manager verwenden.**

```
# clia shelf deactivate Hardware-Adr FRU-ID 1
```

Beispiel:

```
# clia shelf deactivate 0x41 0xfe 1
IPM Sentry Shelf Manager Command Line Interpreter
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info, address: 0x41, FRU ID # 254
    Cached information updated
    Wrote Information to the Shelf FRU
#
```

ANHANG **A**

# Shelf Manager-CLI-Befehle

In diesem Kapitel werden die einzelnen CLI-Befehle beschrieben und die Syntax und Anwendung der verfügbaren Befehle erläutert. Das CLI unterstützt AdvancedTCA-Shelf-Kontexte.

Um die Benutzerfreundlichkeit zu erhöhen, kann auf Schlüsseltypen von Shelf-Komponenten auch wie folgt verwiesen werden. Dies stellt eine Alternative zur Verweisnotation dar, die auf der IPMB-Adresse und dem numerischen FRU-Bezeichner basiert:

- `board` *n* | `b` *n*
- `fan_tray` *n* | `ft` *n*
- `shm 1 | 2`

---

**Hinweis –** Sie können die Notation `shm 1` und `shm 2` verwenden, um auf die redundanten Shelf Manager zuzugreifen, die in der Shelf-FRU in der Adresstabelle beschrieben sind. In diesem Handbuch bezieht sich `shm 1` auf den Shelf Manager mit der numerisch kleineren Hardwareadresse, und `shm 2` bezieht sich auf den Shelf Manager mit der numerisch größeren Hardwareadresse.

---

Bei redundanten Konfigurationen werden nicht alle unten aufgeführten Befehle vom Sicherungs-Shelf Manager unterstützt. Der Sicherungs-Shelf Manager erkennt die folgenden Befehle:

- `debuglevel`
- `localaddress`
- `shmstatus`
- `switchover`

Die meisten Informationsbefehle unterstützen den kurzen und den ausführlichen Ausführungsmodus. Der Unterschied liegt in der Menge der bereitgestellten Informationen. Der kurze Modus ist die Standardeinstellung. Sie wählen den ausführlichen Modus aus, indem Sie in der Befehlszeile die Option `-v` direkt nach dem Befehl und vor den Positionsargumenten verwenden.

In der Befehlssyntax unten sind optionale Elemente in eckige Klammern (`[,]`) gesetzt, und variable Elemente der Befehlszeile (z. B. IPMB-Adresse und FRU-Geräte-ID) sind *kursiv* gedruckt. Parameteralternativen sind durch einen vertikalen Strich (|) gekennzeichnet.

# activate

### Syntax:

`activate` *IPMB-adresse fru-id*
`activate board` *n*
`activate shm` *n*

### Zweck:

Dieser Befehl sendet den IPMI-Befehl `Set FRU Activation (Activate FRU)` an die angegebene FRU. Die FRU wird angegeben, indem die IPMB-Adresse des als Eigentümer fungierenden IPM-Controllers und die FRU-Geräte-ID verwendet wird. Die FRU-Geräte-ID 0 bezeichnet in PICMG 3.0-Kontexten den IPM-Controller.

Im PICMG 3.0-Kontext eignet sich dieser Befehl vor allem für die FRUs, die nicht in der Energieverwaltungstabelle der Shelf-FRU-Informationen aufgeführt sind oder für die das Attribut `Shelf Manager Controlled Activation` auf `FALSE` gesetzt ist. Diese FRUs werden nicht automatisch vom Shelf Manager aktiviert und bleiben im M2-Status (Activation Request = Aktivierungsanforderung). Der Shelf Manager aktiviert andere FRUs automatisch, sobald sie den M2-Status erreichen. Wenn Sie versuchen, eine FRU zu aktivieren, die sich nicht im M2-Status befindet, wird keine Aktion ausgeführt.

### Beispiel:

Richtiges Aktivieren des IPM-Controllers unter der Adresse 9C.

```
# clia activate 9c 0
IPM Sentry Shelf Manager Command Line Interpreter
Command issued via IPMB, status = 0 (0x0)
Command executed successfully
#
```

# alarm

### Syntax:

`alarm [clear|minor|major|critical]`

### Zweck:

Dieser Befehl ermöglicht den Zugriff auf Telco Alarm-Ausgaben. Mithilfe der Parameter `minor`, `major` und `critical` können Benutzer die entsprechende Alarmausgabe festlegen. Diese Aktionen sind kumulativ. Dies bedeutet, dass nach den Befehlen `clia alarm minor` und `clia alarm major` sowohl weniger wichtige (minor) als auch wichtige Alarme (major) festgelegt werden. Die Aktion `clear` löscht diese beiden Arten von Alarmausgaben. Kritische Alarmausgaben (critical) können jedoch nicht gelöscht werden.

Wenn Sie den Befehl ohne Parameter verwenden, wird der Status von Telco Alarm-Ausgaben zurückgegeben.

### Beispiel:

```
# clia alarm
IPM Sentry Shelf Manager Command Line Interpreter
    alarm mask: 0x00
# clia alarm major
IPM Sentry Shelf Manager Command Line Interpreter
Returned completion code: 0
# clia alarm
IPM Sentry Shelf Manager Command Line Interpreter
    alarm mask: 0x02
# clia alarm clear
IPM Sentry Shelf Manager Command Line Interpreter
Returned completion code: 0
# clia alarm
IPM Sentry Shelf Manager Command Line Interpreter
alarm mask: 0x00
```

# board

### Syntax:

`board` [`-v`] [*physikalische_steckplatzadresse*]

### Zweck:

Dieser Befehl und der Befehl `boardreset` unterscheiden sich von den restlichen Befehlen darin, dass sie mit ATCA-Boards arbeiten und Nummern von physikalischen Steckplätzen als Argumente verwenden, anstatt IPM-Controlleradressen und FRU-Geräte-IDs. Der Befehl `board` zeigt Informationen zu den einzelnen IPM-Controllern im Bereich der IPMB-Adressen an, die den ATCA-Steckplätzen zugeordnet sind. Außerdem zeigt er Informationen zu den zusätzlichen FRUs an, die mithilfe dieser Controller gesteuert werden. Die Liste der angezeigten Elemente finden Sie in den Abschnitten „`fru`“ auf Seite 143 und „`ipmc`“ auf Seite 191. Der Bereich der IPMB-Adressen lautet für PICMG 3.0-Systeme 82h bis A0h, in dem Boards über IPM-Controller verfügen.

Die physikalische Adresse muss als Dezimalzahl angegeben werden. Bei PICMG 3.0-Systemen wird die Korrespondenz zwischen den physikalischen Adressen und IPMB-Adressen in den Shelf-FRU-Informationen angegeben. Wenn die Shelf-FRU-Informationen keine Adresstabelle enthalten, wird die folgende Zuordnungstabelle (Zuordnung logischer Steckplatznummern) verwendet.

| Steckplatznummer | IPMB-Adresse |
|---|---|
| 1 | 9A |
| 2 | 96 |
| 3 | 92 |
| 4 | 8E |
| 5 | 8A |
| 6 | 86 |
| 7 | 82 |
| 8 | 84 |
| 9 | 88 |
| 10 | 8C |
| 11 | 90 |

| Steckplatznummer | IPMB-Adresse |
|---|---|
| 12 | 94 |
| 13 | 98 |
| 14 | 9C |

### Beispiele:

Abrufen von Standardinformationen zu allen Boards des Systems (wobei nur die Boards in den physikalischen Steckplätzen 1 und 14 vorhanden sind).

```
# clia board
IPM Sentry Shelf Manager Command Line Interpreter
Physical Slot # 1
82: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
82: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
Physical Slot # 14
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
#
```

Abrufen von ausführlichen Informationen zu einem Board im physikalischen Steckplatz 14.

```
# clia board -v 14
Physical Slot # 14
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID: 0x00, Revision: 0, Firmware: 1.01, IPMI ver 1.5
```

```
    Manufacturer ID: 00315a (PICMG), Product ID: 0000, Auxiliary
Rev: 01ac1014
    Device ID String: "IPM Sentry 6"
    Global Initialization: 0x0, Power State Notification: 0x0,
Device Capabilities: 0x29
    Controller provides Device SDRs
    Supported features: 0x29
        "Sensor Device" "FRU Inventory Device" "IPMB Event
Generator"
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
    Site Type: 0x00, Site Number: 14
    Current Power Level: 0x01, Maximum Power Level: 0x01, Current
Power Consumption: 20.0 Watts
#
```

# boardreset

### Syntax:

`boardreset` *physikalische_steckplatzadresse*

### Zweck:

Dieser Befehl setzt das Board im angegebenen physikalischen Steckplatz zurück, indem der IPMI-Befehl `FRU Control (Cold Reset)` gesendet wird.

Die physikalische Adresse muss als Dezimalzahl angegeben werden. Bei PICMG 3.0-Systemen wird die Korrespondenz zwischen den physikalischen Adressen und IPMB-Adressen in den Shelf-FRU-Informationen angegeben. Wenn die Shelf-FRU-Informationen keine Adresstabelle enthalten, wird die folgende Zuordnungstabelle (Zuordnung logischer Steckplatznummern) verwendet. Die FRU-Geräte-ID ist 0.

| Steckplatznummer | IPMB-Adresse |
|---|---|
| 1 | 9A |
| 2 | 96 |
| 3 | 92 |
| 4 | 8E |

| Steckplatznummer | IPMB-Adresse |
|---|---|
| 5 | 8A |
| 6 | 86 |
| 7 | 82 |
| 8 | 84 |
| 9 | 88 |
| 10 | 8C |
| 11 | 90 |
| 12 | 94 |
| 13 | 98 |
| 14 | 9C |

### Beispiel:

Zurücksetzen des Boards im physikalischen Steckplatz 14 (IPMB-Adresse 9C, FRU 0).

```
# clia boardreset 14
IPM Sentry Shelf Manager Command Line Interpreter
Board 14 reset, status returned 0
#
```

## busres

### Syntax:

`busres` *unterbefehl*

Die folgenden Unterbefehle werden unterstützt:

- `info [`*ressource*`]`
- `release` *ressource*
- `force` *ressource*
- `lock` *ressource*
- `unlock` *ressource*
- `query [-v]` *ressource* `[`*ziel* `[noupdate]]`
- `setowner` *ressource* *ziel*
- `sendbusfree` *ressource* *ziel*

### Zweck:

Dieser Befehl zeigt Informationen zu den Ressourcen mit Bus-geleiteter E-Keying-Verwaltung an und ermöglicht das Ändern dieses Zustands.

Alle Unterbefehle akzeptieren eine Ressourcen-ID als Parameter. Bei der Ressourcen-ID handelt es sich entweder um eine auf 0 basierende Ressourcennummer oder um einen kurzen Ressourcennamen. Die folgenden Ressourcennamen und -nummern sind definiert:

| Nummer | Kurzname | Beschreibung |
|---|---|---|
| 0 | mtb1 | Metalic Test Bus, Paar 1 |
| 1 | mtb2 | Metalic Test Bus, Paar 2 |
| 2 | clk1 | Synch-Uhr, Gruppe 1 |
| 3 | clk2 | Synch-Uhr, Gruppe 2 |
| 4 | clk3 | Synch-Uhr, Gruppe 3 |

In den folgenden Unterabschnitten wird die Syntax des Befehls `busres` für einige wichtige Anwendungsmöglichkeiten beschrieben.

## Anzeigen des Status für Ressourcen mit Bus-geleiteter E-Keying-Verwaltung

### Syntax:

`busres info [`*ressource*`]`

### Zweck:

Dieser Befehl zeigt Informationen zum aktuellen Status der angegebenen Ressource oder aller Ressourcen an, wenn keine Ressourcen-ID angegeben wird.

Der Parameter *ressource* ist die Ressourcen-ID. Die Liste der unterstützten Ressourcen-IDs finden Sie unter „busres“ auf Seite 131.

### Beispiel:

Abrufen von Informationen zum Status des Metalic Test Bus, Paar 2.

```
# clia busres info mtb2
IPM Sentry Shelf Manager Command Line Interpreter
Metalic Test Bus pair 2 (ID 1): Owned by IPMC 0x82, Locked
#
```

## Freigeben einer angegebenen Ressource

### Syntax:

`busres release | force` *ressource*

### Zweck:

Dieser Befehl sendet die Anforderung `Bused Resource Control` an den momentanen Eigentümer der Ressource und weist ihn an, die Ressource freizugeben. Wenn die Befehlssyntax `busres release` *ressource* lautet, wird der Befehl `Bused Resource Control (Release)` gesendet. Wenn die Befehlssyntax `busres force` *ressource* lautet, wird der Befehl `Bused Resource Control (Force)` gesendet. Eine ausführliche Beschreibung dieser ATCA-Befehle finden Sie in der Spezifikation PICMG 3.0 R1.0 im Abschnitt 3.7.3.4.

Der Parameter *ressource* ist die Ressourcen-ID. Die Liste der unterstützten Ressourcen-IDs finden Sie unter „`busres`" auf Seite 131.

### Beispiel:

Erzwingen der Freigabe von Metalic Test Bus, Paar 2 vom momentanen Eigentümer.

```
# clia busres force mtb2
IPM Sentry Shelf Manager Command Line Interpreter
Force operation succeeded
#
```

# Sperren/Entsperren der angegebenen Ressource

### Syntax:

`busres lock | unlock` *ressource*

### Zweck:

Dieser Befehl sperrt (`busres lock` *ressource*) oder entsperrt (`busres unlock` *ressource*) die angegebene Ressource. Ist die Ressource gesperrt, antwortet der Shelf Manager mit dem Status **Deny**, wenn ein anderer IPM-Controller den Befehl `Bused Resource Control (Request)` an den Shelf Manager sendet. Ist die Ressource entsperrt, antwortet der Shelf Manager mit dem Status **Busy** und sendet den Befehl `Bused Resource Control (Release)` an den momentanen Eigentümer, wenn ein anderer IPM-Controller den Befehl `Bused Resource Control (Request)` an den Shelf Manager sendet. Wenn der momentane Eigentümer die Ressource freigibt, wird diese Ressource bei der nächsten Anforderung für den Anfordernden bereitgestellt.

Es können nur Ressourcen gesperrt werden, deren Eigentümer ein IPM-Controller ist. Sobald der momentane Eigentümer die Ressource freigibt, wird auch die Sperre dieser Ressource entfernt.

Der Parameter *ressource* ist die Ressourcen-ID. Die Liste der unterstützten Ressourcen-IDs finden Sie unter „`busres`“ auf Seite 131.

### Beispiel:

Sperren der Synch-Uhr, Gruppe 3.

```
# clia busres lock clk3
IPM Sentry Shelf Manager Command Line Interpreter
Lock operation succeeded
#
```

# Senden des Befehls `Bused Resource Control (Query)`

### Syntax:

busres [-v] query *ressource* [*ziel* [noupdate]]

### Zweck:

Dieser Befehl sendet die Anforderung `Bused Resource Control (Query)` an den angegebenen IPM-Controller. Wenn der IPM-Controller in der Befehlszeile nicht angegeben ist, wird die Anforderung an den momentanen Eigentümer der Ressource gesendet. Nach dem Empfang der Antwort werden in der Ressourcentabelle die entsprechenden Änderungen vorgenommen (z. B. wenn der IPM-Controller, der als momentaner Eigentümer angesehen wird, mit dem Status **No Control** antwortet, wird die Tabelle entsprechend geändert), es sei denn, das Flag `noupdate` ist gesetzt. Wenn dieses Flag in der Befehlszeile übergeben wird, werden aufgrund der empfangenen Informationen keine Änderungen an der Ressourcentabelle vorgenommen.

Der Parameter *ressource* ist die Ressourcen-ID. Die Liste der unterstützten Ressourcen-IDs finden Sie unter „`busres`“ auf Seite 131.

Der Parameter *ziel* gibt die IPMB-Adresse des IPM-Controllers an, an den die Anforderung gesendet wird.

Das Flag `noupdate` zeigt (falls vorhanden) an, dass die Informationen, die als Antwort auf die Query-Anforderung empfangen werden, nicht zum Aktualisieren der Ressourcentabelle verwendet werden.

In der aktuellen Shelf Manager-Version werden keine zusätzlichen Informationen bereitgestellt, wenn Sie das Flag `-v` angeben.

### Beispiel:

Senden einer Abfrage für „Metalic Test Bus, Paar 1“ an den IPM-Controller mit der Adresse 0x82. Keine Aktualisierung der Ressourcentabelle aufgrund der als Antwort empfangenen Informationen.

```
# clia busres query mtb1 0x82 noupdate
IPM Sentry Shelf Manager Command Line Interpreter
No Control: IPMC 0x82 is not the owner of resource 0
#
```

# Festlegen des Eigentümers für die Ressource

### Syntax:

`busres setowner` *ressource ziel*

### Zweck:

**Achtung –** Dieser Befehl ist für erfahrene Benutzer gedacht und sollte bei Bedarf entsprechend verwendet werden.

Dieser Befehl legt den Eigentümer der angegebenen Ressource direkt in der Ressourcentabelle fest. Er sendet keine Bused Resource Control-Befehle. Dies ist auch dann nicht der Fall, wenn die Ressource vor der Ausführung des Befehls einen anderen Eigentümer hatte. Es handelt sich um einen Low-Level-Befehl, der nur zu Test- und Wiederherstellungszwecken eingesetzt werden darf.

Der Parameter *ressource* ist die Ressourcen-ID. Die Liste der unterstützten Ressourcen-IDs finden Sie unter „busres“ auf Seite 131.

Der Parameter *ziel* gibt die IPMB-Adresse des IPM-Controllers an, der als Eigentümer der Ressource festgelegt ist. Verwenden Sie 0 als IPMB-Adresse, um anzugeben, dass für die Ressource kein IPM-Controller als Eigentümer fungiert.

### Beispiel:

Festlegen von Board 1 als neuen Eigentümer für „Metalic Test Bus, Paar 1“.

```
# clia busres setowner mtb1 board 1
IPM Sentry Shelf Manager Command Line Interpreter
New owner is set successfully
#
```

# Senden des Befehls `Bused Resource Control (Bus Free)`

**Syntax:**

`busres sendbusfree` *ressource ziel*

**Zweck:**

**Achtung –** Dieser Befehl ist für erfahrene Benutzer gedacht und sollte bei Bedarf entsprechend verwendet werden.

Dieser Befehl sendet die Anforderung `Bused Resource Control (Bus Free)` an den angegebenen IPM-Controller. Für die Ressource wird keine Operation ausgeführt, bevor die Anforderung gesendet wird. Dies gilt auch, wenn ein anderer IPM-Controller der Eigentümer ist. Die Ressourcentabelle wird jedoch aufgrund der Antwort auf diese Anforderung aktualisiert. Wenn der IPM-Controller also die Rolle als Eigentümer der Ressource akzeptiert, wird er in der Tabelle als neuer Eigentümer aufgeführt. Es handelt sich um einen Low-Level-Befehl, der nur zu Test- und Wiederherstellungszwecken eingesetzt werden darf.

Der Parameter *ressource* ist die Ressourcen-ID. Die Liste der unterstützten Ressourcen-IDs finden Sie unter „`busres`“ auf Seite 131.

Der Parameter *ziel* gibt die IPMB-Adresse des IPM-Controllers an, an den die Anforderung gesendet wird. Verwenden Sie 0 als IPMB-Adresse, um anzugeben, dass für die Ressource kein IPM-Controller als Eigentümer fungiert.

**Beispiel:**

Senden einer Bus Free-Anforderung für „Metalic Test Bus, Paar 1“ an den IPM-Controller mit der Adresse 0x82.

```
# clia busres sendbusfree mtb1 0x82
IPM Sentry Shelf Manager Command Line Interpreter
IPMC rejected ownership of the resource
#
```

# console

### Syntax:

`console` *steckplatznummer*

### Zweck:

Dieser Befehl richtet eine Konsolenterminalsitzung für das Node Board im angegebenen physikalischen Steckplatz ein. Der Shelf Manager lässt eine Konsolensitzung pro Node Board zu. Die gültigen Steckplatznummern sind 1 bis 6 und 9 bis 14.

---

**Hinweis –** Die primäre (bzw. obere) Shelf-Verwaltungskarteneinheit muss die **aktive** Shelf-Verwaltungskarte sein, um die Konsolenfunktion verwenden zu können. Außerdem muss in Steckplatz 7 der Mittelplatine des Netra CT 900 Servers eine Switch-Karte installiert sein.

---

Nachdem eine Konsolensitzung für ein Node Board eingerichtet wurde, können Sie Systemadministrationsbefehle ausführen, z. B. `passwd`, Status- und Fehlermeldungen lesen oder das Board im jeweiligen Steckplatz anhalten.

---

**Hinweis –** Wenn an den seriellen Port des Node Boards ein Konsolen- oder ein serielles Kabel angeschlossen ist, erfolgt die Konsolenausgabe auf der per Kabel verbundenen Konsole, anstatt über die Konsolensitzung auf dem ShMM. Dies ist auch der Fall, wenn die Konsolensitzung des ShMM beim Anschließen des Kabels aktiv war.

---

Wenn Sie die Verbindung zur aktuellen Konsolensitzung abbrechen möchten, geben Sie `~q` oder `~.` ein (Tilde Punkt).

### Beispiel:

Starten einer Konsolensitzung auf dem Node Board im physikalischen Steckplatz 4.

```
# clia console 4
eingabeaufforderung
```

# deactivate

### Syntax:

deactivate *IPMB-adresse fru-id*
deactivate board *n*
deactivate shm *n*

### Zweck:

Dieser Befehl sendet den IPMI-Befehl `Set FRU Activation (Deactivate FRU)` an die angegebene FRU. Die FRU wird angegeben, indem die IPMB-Adresse des als Eigentümer fungierenden IPM-Controllers und die FRU-Geräte-ID verwendet wird. Die FRU-Geräte-ID 0 bezeichnet in PICMG 3.0-Kontexten den IPM-Controller.

### Beispiel:

Richtiges Deaktivieren des IPM-Controllers bei Adresse 9C.

```
# clia deactivate 9c 0
IPM Sentry Shelf Manager Command Line Interpreter
Command issued via IPMB, status = 0 (0x0)
    Command executed successfully
#
```

# debuglevel

### Syntax:

debuglevel [*neuer_wert*]

### Zweck:

Dieser Befehl zeigt für den IPM Sentry Shelf Manager die aktuelle Debug-Ebene an oder setzt sie auf einen neuen Wert, wenn ein neuer Wert angegeben ist.

Die Debug-Ebene ist eine hexadezimale Zahl im Bereich 0x0000 bis 0x00FF, die als Bitmaske behandelt wird. Jedes Bit der Maske aktiviert (wenn es gesetzt ist) eine Debug-Ausgabe eines bestimmten Typs:

- 0x0001 – Fehlermeldungen

- 0x0002 – Warnmeldungen
- 0x0004 – Informationsmeldungen
- 0x0008 – Ausführliche Informationsmeldungen
- 0x0010 – Verfolgungsmeldungen
- 0x0020 – Ausführliche Verfolgungsmeldungen
- 0x0040 – Meldungen, die für wichtige Befehle angezeigt werden, die während der Initialisierung an die IPM-Controller gesendet werden
- 0x0080 – Ausführliche Meldungen zur Einrichtung und Freigabe von internen Sperren

Die standardmäßige Debug-Ebene für den Shelf Manager ist 0x0007. Sie können diesen Wert beim Starten des Shelf Managers jedoch außer Kraft setzen, indem Sie in der Befehlszeile die Option `-v` verwenden. Mithilfe des CLI können Sie die Debug-Ebene auch während der Laufzeit ändern.

### Beispiel:

Abrufen der aktuellen Debug-Ebene und Einstellen auf 0x001F.

```
# clia debuglevel
IPM Sentry Shelf Manager Command Line Interpreter
Debug Mask is 0x0007
# clia debuglevel 1f
IPM Sentry Shelf Manager Command Line Interpreter
# clia debuglevel
IPM Sentry Shelf Manager Command Line Interpreter
Debug Mask is 0x001f
```

# exit | quit

### Syntax:

```
exit
quit
```

### Zweck:

Der Befehl `exit` oder `quit` beendet den interaktiven CLI-Modus (der aktiviert wird, indem Sie `clia` ohne Parameter verwenden).

### Beispiel:

```
# exit
#
```

# fans

### Syntax:

`fans [-v] [`*IPMB-adresse* `[`*FRU-geräte-ID*`]]`
`fans fan_tray` *n*

### Zweck:

Dieser Befehl zeigt Informationen zu den angegebenen Lüfter-FRUs an. Wenn Sie die FRU-Geräte-ID nicht angeben, zeigt der Befehl Informationen zu allen Lüfter-FRUs an, die unter der angegebenen Adresse vom IPM-Controller gesteuert werden. Wenn auch die IPMB-Adresse fehlt, zeigt der Befehl Informationen zu allen Lüfter-FRUs an, die dem Shelf Manager bekannt sind. Die folgenden Informationen werden angezeigt:

- IPMB-Adresse und FRU-Geräte-ID
- Minimale Geschwindigkeitseinstellung
- Maximale Geschwindigkeitseinstellung
- Maximale dauerhafte Geschwindigkeitseinstellung
- Aktuelle Einstellung (Einstellung nach dem Außerkraftsetzen und lokale Einstellung, falls jeweils vorhanden)

### Beispiel:

Abrufen der Lüfterinformationen zu allen Lüfter-FRUs unter der IPMB-Adresse 20.

```
# clia fans 20
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 3
 Current Level: 6
 Minimum Speed Level: 0, Maximum Speed Level: 15
20: FRU # 4
 Current Level: 255 "Automatic"
 Minimum Speed Level: 0, Maximum Speed Level: 15
20: FRU # 5
 Current Level: 255 "Automatic"
 Minimum Speed Level: 0, Maximum Speed Level: 15
#
```

## flashupdate

### Syntax:

`flashupdate` *steckplatznummer* `-s` *server-ip-adresse* `-f` *fw-abbild-pfadname*

### Zweck:

Dieser Befehl wird verwendet, um die Systemfirmware des Netra CP3060 Boards mit dem Firmware-Abbild von einem Speicherort zu aktualisieren, den Sie angeben. Dieser Befehl ist nur für Netra CP3060 Boards gültig, die im Netra CT 900 Server installiert sind. Zu der Systemfirmware des Netra CP3060 Boards, die aktualisiert wird, gehört ALOM-CMT-, Hypervisor-, OBP-, Post- und VBSC-Firmware.

---

**Hinweis –** Die Links zur Download-Website für die Netra CP3060-Firmware finden Sie unter: `http://www.sun.com/downloads/`

---

Sie müssen Folgendes kennen, um diesen Befehl verwenden zu können:

- IP-Adresse des FTP-Servers, von dem Sie das Firmware-Abbild herunterladen möchten
- Benutzername und Passwort des FTP-Servers zum Eingeben an den Eingabeaufforderungen
- Pfad, unter dem das Abbild gespeichert ist

*steckplatznummer* steht für die Steckplatznummer des Netra CP3060 Boards, das Argument **-s** *server-ip-adresse* gibt die IP-Adresse des Servers an, von dem das Firmware-Abbild heruntergeladen wird, und unter `-f` *fw-abbild-pfadname* ist der vollständige Pfadname angegeben, unter dem das Firmware-Abbild abgelegt ist.

### Beispiele:

Herunterladen und Aktualisieren der Systemfirmware für ein Netra CP3060 Board. Beachten Sie, dass die Ausführung dieses Vorgangs mehrere Minuten in Anspruch nehmen kann. Setzen Sie das Board nach der erfolgreichen Ausführung zurück, indem Sie den Befehl `boardreset` verwenden.

```
# clia flashupdate 2 -s 123.45.67.89
  -f /sysfw/System_Firmware-6_2_5-Netra_CP3060.bin
Username: username
Password: *******
................................................................
................................................................
.....................................
Update complete. Reset device to use new software.

# clia boardreset slot-number
```

# fru

### Syntax:

`fru [-v] [`*adresse* `[id=`*fru_id* `| type=`*standorttyp*`]] | [type=`*standorttyp* `[/`*standortnummer*`]]`<br>
`fru board` *n*<br>
`fru shm` *n*<br>
`fru fan_tray` *n*

### Zweck:

Dieser Befehl zeigt Informationen zu einer bestimmten FRU an. Wenn Sie die FRU-Geräte-ID nicht angeben, zeigt der Befehl Informationen zu allen FRUs an, die unter der angegebenen Adresse vom IPM-Controller gesteuert werden. Wenn auch die IPMB-Adresse fehlt, zeigt der Befehl Informationen zu allen FRUs an, die dem Shelf Manager bekannt sind.

Außerdem können Sie über die Standorttypen FRUs auswählen. Sie müssen den Standorttyp in den Befehlsparametern hexadezimal angeben. Die Zuordnungen zwischen FRUs und ihren Standorttypen sind in den Shelf-FRU-Informationen gespeichert. Standorttypen sind in der Spezifikation PICMG 3.0 folgendermaßen definiert:

- 00h – AdvancedTCA Board
- 01h – Stromversorgungsmodul
- 02h – Shelf-FRU-Informationen
- 03h – Dedizierter ShMC
- 04h – Lüfterelement
- 05h – Lüfterfilterelement
- 06h – Alarm
- 07h – AdvancedTCA-Modul (Mezzanine)
- 08h – PMC
- 09h – Rear Transition Card
- C0h bis CFh – Vom OEM definiert
- Alle anderen Werte sind reserviert.

Im Standardmodus werden für die FRU die folgenden Informationen angezeigt:

- IPMB-Adresse und FRU-Geräte-ID
- Entitäts-ID, Entitätsinstanz
- Standorttyp und -nummer (falls bekannt)
- Aktueller Hot-Swap-Status, vorheriger Hot-Swap-Status und Ursache der letzten Statusänderung der FRU. Die Hot-Swap-Statustypen M0 bis M7 sind in der Spezifikation PICMG 3.0 folgendermaßen definiert:
  - M0 – Nicht installiert
  - M1 – Inaktiv
  - M2 – Aktivierungsanforderung
  - M3 – Aktivierung wird durchgeführt
  - M4 – FRU aktiv
  - M5 – Deaktivierungsanforderung
  - M6 – Deaktivierung wird durchgeführt
  - M7 – Kommunikation unterbrochen

Die folgenden Informationen werden für die FRU nur im ausführlichen Modus angezeigt:

- FRU-Gerätetyp, Gerätetyp-Modifizierer (nur für `FRU-device-ID != 0`). Diese Informationen stammen aus dem FRU Sensor Data Record (SDR) und entsprechen Abschnitt 37.12 der IPMI-Spezifikation.
- Geräte-ID-Zeichenfolge aus dem FRU SDR
- Aktuelle FRU-Leistungsstufe und maximale FRU-Leistungsstufe; aktuell zugeordnete Leistungsaufnahme in Watt

**Beispiele:**

Abrufen von Standardinformationen zu allen FRUs unter der Adresse 9C.

```
# clia fru 9c 0
IPM Sentry Shelf Manager Command Line Interpreter
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
#
```

Abrufen von ausführlichen Informationen zu allen FRUs unter der Adresse 9C.

```
# clia fru -v 9c 0
IPM Sentry Shelf Manager Command Line Interpreter
9c: FRU # 0
    Entity: (0xd0, 0x0)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID String: "IPM Sentry 6"
    Site Type: 0x00, Site Number: 14
    Current Power Level: 0x01, Maximum Power Level: 0x01, Current
Power Consumption: 20.0 Watts
#
```

Abrufen von ausführlichen Informationen zu FRU 1 unter der Adresse 20.

```
# clia fru -v 20 id=1
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1
    Entity: (0x1, 0x1)
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device Type: "FRU Inventory Device behind management
controller" (0x10), Modifier 0x0
    Device ID String: "IPM Sentry 1.1"
    Current Power Level: UNKNOWN, Maximum Power Level: UNKNOWN,
Current Power Consumption: UNKNOWN
#
```

# frucontrol

### Syntax:

frucontrol *IPMB-adresse fru-id option*

frucontrol board *n option*
frucontrol shm *n option*
frucontrol fan_tray *n option*

### Zweck:

Dieser Befehl sendet den FRU-Steuerbefehl an die angegebene FRU, um die angegebene Operation für die FRU-Datenlast auszuführen. Die FRU wird angegeben, indem die IPMB-Adresse des als Eigentümer fungierenden IPM-Controllers und die FRU-Geräte-ID verwendet wird. Die FRU-Geräte-ID 0 bezeichnet in PICMG 3.0-Kontexten den IPM-Controller.

Der Parameter *option* gibt an, welche Option des FRU-Steuerbefehls verwendet wird. Sie können die folgenden symbolischen Werte angeben:

- `cold_reset` (abgekürzt als `cr`) – Kaltstart der FRU-Datenlast durchführen
- `warm_reset` (abgekürzt als `wr`) – Warmstart der FRU-Datenlast durchführen
- `graceful_reboot` (abgekürzt als `gr`) – Sanften Neustart der FRU-Datenlast durchführen
- `diagnostic_interrupt` (abgekürzt als `di`) – Diagnoseunterbrechung durchführen

### Beispiel:

Ausgeben eines Kaltstart-Befehls an FRU 0 unter der IPMB-Adresse 9C.

```
# clia frucontrol 9c 0 cr
IPM Sentry Shelf Manager Command Line Interpreter
    FRU Control: Controller 0x9c, FRU ID # 0, command 0x00, status
0(0x0)
     Command executed successfully
#
```

# frudata

### Syntax:

frudata [*adresse* [*fru_id* [*blockabstand*]]]
frudata *adresse fru_id byte-abstand byte 1* [*byte2* ... [*byte 16*] ...]

Sie können *adresse fru_id* durch Folgendes ersetzen:

board *n*
shm *n*
fan_tray *n*

### Zweck:

Dieser Befehl gewährt den Zugriff auf die FRU-Informationen in Rohform. Je nach Format des Befehls wird er verwendet, um die FRU-Informationen zu lesen oder zu schreiben.

Im Leseformat verwendet der Befehl eine optionale 32-Byte-Blocknummer.

Im Schreibformat ist ein Parameter für den Byte-Abstand erforderlich. Benutzer können bis zu 65 535 Byte an FRU-Informationen ändern.

frudataw und frudatar sind Varianten des Befehls frudata. frudataw ermöglicht es Benutzern, eine Datei aus dem ShMM-Flash-Dateisystem in den FRU-Informationsspeicher einer bestimmten FRU im Shelf zu schreiben (siehe „frudatar" auf Seite 149). Mithilfe von frudatar können Benutzer den Inhalt des FRU-Informationsspeichers für eine bestimmte FRU in eine Datei des ShMM-Flash-Dateisystems übertragen (siehe „frudatar" auf Seite 149).

### Beispiele:

Anzeigen der FRU-Standarddaten für alle FRUs.

```
# clia frudata
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 0    Failure status: 203 (0xcb)
    Requested data not present
20: FRU # 1 Raw FRU Info Data
    FRU Info size: 529
20: FRU # 2    Failure status: 203 (0xcb)
    Requested data not present
82: FRU # 0 Raw FRU Info Data
    FRU Info size: 160
```

```
9c: FRU # 0 Raw FRU Info Data
    FRU Info size: 160
fc: FRU # 0 Raw FRU Info Data
    Requested data not present
. . .
#
```

Dieses Beispiel zeigt, wie Sie FRU-Daten anzeigen und Daten in eine FRU schreiben.

```
# clia frudata 20 1 0
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1 Block # 0 Raw FRU Info Data
    FRU Info size: 529
 01 00 01 05  0E 18 00 D3  01 04 01 02  55 AA 83 55
 AA 55 C1 00  00 00 00 00  00 00 00 00  00 00 00 00
#
# clia frudata 20 1 1 0xfc 0xfe
IPM Sentry Shelf Manager Command Line Interpreter
Writing 2 bytes to IPM 0x20, FRU # 1, offset: 1, status = 0(0x0)
#
# clia frudata 20 1 0
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1 Block # 0 Raw FRU Info Data
    FRU Info size: 529
 01 FC FE 05  0E 18 00 D3  01 04 01 02  55 AA 83 55
 AA 55 C1 00  00 00 00 00  00 00 00 00  00 00 00 00
#
# clia frudata 20 1 1 0 1
IPM Sentry Shelf Manager Command Line Interpreter
Writing 2 bytes to IPM 0x20, FRU # 1, offset: 1, status = 0(0x0)
#
# clia frudata 20 1 0
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1 Block # 0 Raw FRU Info Data
    FRU Info size: 529
 01 00 01 05  0E 18 00 D3  01 04 01 02  55 AA 83 55
 AA 55 C1 00  00 00 00 00  00 00 00 00  00 00 00 00
#
```

# frudatar

### Syntax:

frudatar *adresse fru_id dateiname*
frudatar *adresse fru_id byte_abstand byte 1* [*byte2* ... [*byte 16*] ...]

Sie können *adresse fru_id* durch Folgendes ersetzen:

board *n*
shm *n*
fan_tray *n*

### Zweck:

Dieser Befehl liest FRU-Informationen aus der angegebenen FRU und speichert sie in einer Datei im ShMM-Flash-Dateisystem (es werden also FRU-Informationen aus der angegebenen FRU in eine Flash-Datei hochgeladen). Der Parameter *dateiname* gibt den Pfad zur Zieldatei an. Die Anzahl der Byte, die aus der FRU gelesen und in die Zieldatei geschrieben werden, entspricht der Anzahl an Byte, die für die angegebene FRU als Antwort auf den IPMI-Befehl `Get FRU Inventory Area Info` zurückgegeben werden.

### Beispiel:

Lesen von FRU-Daten für eine bestimmte FRU und Speichern der Daten in der angegebenen Datei.

```
# clia frudatar 20 2 /var/tmp/20.2.bin
IPM Sentry Shelf Manager Command Line Interpreter

20: FRU # 2 Raw FRU Info Data
 FRU Info size: 176
01 00 00 01  09 00 00 F5  01 08 19 84  C0 42 C7 53
63 68 72 6F  66 66 D9 53  68 4D 4D 2D  41 43 42 2D
46 43 20 53  68 65 6C 66  20 4D 61 6E  61 67 65 72
86 10 04 41  10 14 01 89  D2 04 65 58  13 51 17 00
00 C0 C1 00  00 00 00 EA  01 0D 19 C7  53 63 68 72
6F 66 66 DD  46 61 6E 20  43 6F 6E 74  72 6F 6C 6C
65 72 20 6F  6E 20 53 68  4D 4D 2D 41  43 42 2D 46
43 89 D2 04  65 58 13 51  17 00 00 C9  52 65 76 2E
20 31 2E 30  30 86 10 04  41 10 14 01  C0 DF 2F 76
```

```
61 72 2F 6E  76 64 61 74  61 2F 66 61  6E 2D 66 72
75 2D 69 6E  66 6F 72 6D  61 74 69 6F  6E C1 00 26
#
```

# frudataw

### Syntax:

`frudataw` *adresse fru_id dateiname*
`frudataw` *adresse fru_id byte-abstand byte 1* [*byte2 … [byte 16*] …]

Sie können *adresse fru_id* durch Folgendes ersetzen:

`board` *n*
`shm` *n*
`fan_tray` *n*

### Zweck:

Dieser Befehl lädt aus einer Datei im ShMM-Flash-Dateisystem FRU-Informationen in die angegebene FRU herunter. Die Datei enthält das binäre Rohdatenabbild der FRU-Informationen. Der Parameter *dateiname* gibt den Pfad zur Quelldatei an.

### Beispiel:

Schreiben von FRU-Daten aus einer Datei in die angegebene FRU.

```
# clia frudataw 20 2 /var/tmp/20.2.bin
IPM Sentry Shelf Manager Command Line Interpreter

 Writing 16 bytes to IPM 0x20, FRU # 2, offset: 0, status = 0(0x0)
 Writing 16 bytes to IPM 0x20, FRU # 2, offset: 16, status = 0(0x0)
 Writing 16 bytes to IPM 0x20, FRU # 2, offset: 32, status = 0(0x0)
 Writing 16 bytes to IPM 0x20, FRU # 2, offset: 48, status = 0(0x0)
 Writing 16 bytes to IPM 0x20, FRU # 2, offset: 64, status = 0(0x0)
 Writing 16 bytes to IPM 0x20, FRU # 2, offset: 80, status = 0(0x0)
 Writing 16 bytes to IPM 0x20, FRU # 2, offset: 96, status = 0(0x0)
 Writing 16 bytes to IPM 0x20, FRU # 2, offset: 112, status = 0(0x0)
 Writing 16 bytes to IPM 0x20, FRU # 2, offset: 128, status = 0(0x0)
 Writing 16 bytes to IPM 0x20, FRU # 2, offset: 144, status = 0(0x0)
 Writing 16 bytes to IPM 0x20, FRU # 2, offset: 160, status = 0(0x0)
 File "/var/tmp/20.2.orig.bin" has been written to the FRU 20#2
#
```

# fruinfo

### Syntax:

`fruinfo [-v] [-x]` *adresse fru_id*

Sie können *adresse fru_id* durch Folgendes ersetzen:

board *n*
shm *n*
fan_tray *n*

### Zweck:

Dieser Befehl zeigt FRU-Informationen in einem benutzerfreundlichen Format an.

### Beispiel:

Anzeigen von FRU-Informationen für eine bestimmte FRU.

```
# clia fruinfo 20 1
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 1, FRU Info
Common Header:    Format Version = 1
Chassis Info Area:
    Version   = 1
    Chassis Type              = (1)
    Chassis Part Number       = 0x55 0xAA
    Chassis Serial Number     = 5I:5
Board Info Area:
    Version     = 1
Mfg Date/Time            = Jun 16 15:37:00 2011 (8129737 minutes
since 1996)
Board Manufacturer       = Pigeon Point Systems
Board Serial Number      = PPS0000000
    Board Part Number         = A
    FRU Programmer File ID    =
Product Info Area:
    Version     = 1
    Language Code             = 25
    Manufacturer Name         = Pigeon Point Systems
    Product Name              = Shelf Manager
    Product Part / Model#     = 000000
    Product Version           = Rev. 1.00
```

```
    Product Serial Number    = PPS0000000
    Asset Tag                =
    FRU Programmer File ID   =
Multi Record Area:
    Record Type              = Management Access Record
        Version = 2
    Sub-Record Type: Component Name (0x05)
    PICMG Address Table Record (ID=0x10)
        Version = 1
    PICMG Backplane Point-to-Point Connectivity Record (ID=0x04)
        Version = 0
    PICMG Shelf Power Distribution Record (ID=0x11)
        Version = 0
    PICMG Shelf Activation And Power Management Record (ID=0x12)
        Version = 0
#
```

# getfanlevel

### Syntax:

`getfanlevel` *IPMB-adresse fru-id*
`getfanlevel fan_tray` *n*

### Zweck:

Dieser Befehl zeigt die aktuelle Einstellung des Lüfters an, der von der in den Befehlsparametern angegebenen FRU gesteuert wird.

### Beispiel:

Abrufen der Lüftereinstellung für den Lüfter von FRU #2 unter der IPMB-Adresse 0x20.

```
# clia getfanlevel 20 2
IPM Sentry Shelf Manager Command Line Interpreter
20: FRU # 2 Override Fan Level: 1, Local Fan Level: 255
#
```

# getfruledstate

### Syntax:

getfruledstate [-v] [*IPMB-adresse* state [*fru_id* [*LED_ID* | ALL]]]

### Zweck:

Dieser Befehl zeigt den aktuellen Status der FRU-LED für alle Steuereinstellungen an, die für die LEDs aktiviert sind. Im ausführlichen Modus werden außerdem Informationen zu den Farben angezeigt, die von den LEDs unterstützt werden.

Die Informationen können für eine bestimmte LED oder für alle LEDs einer FRU angezeigt werden. Sie können die IPMB-Adresse und die FRU-ID der Ziel-LED auslassen. Wenn die FRU-ID fehlt, werden Informationen zu allen LEDs von allen FRUs des jeweiligen IPM-Controllers angezeigt. Wenn auch die IPMB-Adresse fehlt, werden Informationen zu allen bekannten LEDs des Shelfs angezeigt.

### Beispiele:

Anzeigen des LED-Status für alle LEDs des IPM-Controllers unter der IPMB-Adresse FCh.

```
# clia getfruledstate fc
IPM Sentry Shelf Manager Command Line Interpreter

fc: FRU # 0, Led # 0 ("BLUE LED"):
    Local Control LED State: LED OFF

fc: FRU # 0, Led # 1 ("LED 1"):
    Local Control LED State: LED OFF

fc: FRU # 0, Led # 2 ("LED 2"):
    Local Control LED State: LED OFF

fc: FRU # 0, Led # 3 ("LED 3"):
    Local Control LED State: LED OFF

fc: FRU # 0, Led # 4 ("Application Specific LED# 1"):
    Local Control LED State: LED ON, color: GREEN
```

Anzeigen des LED-Status für alle LEDs des IPM-Controllers unter der IPMB-Adresse FCh.

```
# clia getfruledstate -v  FC
IPM Sentry Shelf Manager Command Line Interpreter

fc: FRU # 0, Led # 0 ("BLUE LED"):
    Local Control LED State: LED OFF
    LED’s color capabilities:
        Colors supported(0x02): BLUE
        Default LED Color in Local Control State(0x01): BLUE
        Default LED Color in Override State(0x01): BLUE

fc: FRU # 0, Led # 1 ("LED 1"):
    Local Control LED State: LED OFF
    LED’s color capabilities:
        Colors supported(0x0C): RED GREEN
        Default LED Color in Local Control State(0x03): GREEN
        Default LED Color in Override State(0x03): GREEN

fc: FRU # 0, Led # 2 ("LED 2"):
    Local Control LED State: LED OFF
    LED’s color capabilities:
        Colors supported(0x0C): RED GREEN
        Default LED Color in Local Control State(0x03): GREEN
        Default LED Color in Override State(0x03): GREEN

fc: FRU # 0, Led # 3 ("LED 3"):
    Local Control LED State: LED OFF
    LED’s color capabilities:
        Colors supported(0x0C): RED GREEN
        Default LED Color in Local Control State(0x02): RED
        Default LED Color in Override State(0x02): RED

fc: FRU # 0, Led # 4 ("Application Specific LED# 1"):
    Local Control LED State: LED ON, color: GREEN
    LED’s color capabilities:
        Colors supported(0x0C): RED GREEN
        Default LED Color in Local Control State(0x02): RED
        Default LED Color in Override State(0x02): RED
```

Anzeigen des LED-Status für FRU #0 des IPM-Controllers unter der IPMB-Adresse 20h.

```
# clia getfruledstate   20 0
IPM Sentry Shelf Manager Command Line Interpreter

20: FRU # 0, Led # 0 ("BLUE LED"):
    Local Control LED State: LED ON, color: BLUE

20: FRU # 0, Led # 1 ("LED 1"):
    Local Control LED State: LED OFF
```

Anzeigen des LED-Status für LED #1 von FRU #0 des IPM-Controllers unter der IPMB-Adresse 20h.

```
# clia getfruledstate  -v  20 0 1
IPM Sentry Shelf Manager Command Line Interpreter

20: FRU # 0, Led # 1 ("LED 1"):
    Local Control LED State: LED OFF
    LED's color capabilities:
        Colors supported(0x04): RED
        Default LED Color in Local Control State(0x02): RED
        Default LED Color in Override State(0x02): RED
```

# gethysteresis

### Syntax:

`gethysteresis [`*IPMB-adresse* `[[`*lun*`: ]`*sensor-id* | *sensorname*`]]`

### Zweck:

Dieser Befehl zeigt die aktuellen Hysteresewerte für den bzw. die angegebenen Sensoren an. Die Sensoren müssen über einen Schwellenwert verfügen. Es werden sowohl Rohwerte als auch verarbeitete Werte angezeigt.

Mithilfe dieses Befehls können Benutzer die Sensornummer mit der Nummer der logischen Einheit (Logical Unit Number, LUN) qualifizieren, wenn der Zielcontroller Sensoren unter mehreren LU-Nummern unterstützt. Wenn die LU-Nummer fehlt, werden die aktuellen Hysteresewerte für alle Sensoren mit der angegebenen Sensornummer angezeigt. *lun* kann den Wert 0, 1 oder 3 haben. (LUN 2 ist reserviert.) Sensornamen sind nicht durch LU-Nummern qualifiziert, da

vorausgesetzt wird, dass Sensornamen innerhalb eines Controllers in der Regel eindeutig sind. Wenn in einem Controller jedoch mehrere Sensoren mit demselben Namen vorhanden sind, werden zu allen Sensoren Informationen angezeigt. Wenn der Wert für *IPMB-adresse* fehlt, werden die aktuellen Hystereseeinstellungen für alle Sensoren unter der angegebenen IPMB-Adresse angezeigt.

### Beispiel:

Anzeigen der Hysteresewerte für Sensor #2 des IPM-Controllers unter der IPMB-Adresse FCh.

```
# clia gethysteresis FC 2
IPM Sentry Shelf Manager Command Line Interpreter

fc: LUN: 0, Sensor # 2 ("lm75 temp")
    Type: Threshold (0x01), "Temperature"(0x01)
        Positive hysteresis, Raw data: 0x00   Processed data:
0.00000 degrees C
        Negative hysteresis, Raw data: 0x00   Processed data:
0.00000 degrees C
```

# getipmbstate

### Syntax:

`getipmbstate` *IPMB-adresse* `[`*link*`]` (in sternförmigen IPMB-0-Umgebungen)
`getipmbstate` *IPMB-adresse* (in Bus-geleiteten IPMB-0-Umgebungen)

### Zweck:

Dieser Befehl zeigt den aktuellen Status von IPMB-0 auf dem IPM-Zielcontroller an. Der Status wird anhand der Sensordaten bestimmt, die vom IPMB-Linksensor auf dem Ziel-IPMC (Sensortyp F1) bereitgestellt werden. Es werden Informationen zu Bus A und Bus B angezeigt.

Der Befehl verhält sich für Bus-geleitete Umgebungen und sternförmige Umgebungen unterschiedlich. Wenn Sie eine Bus-geleitete Umgebung verwenden oder wenn es sich beim Ziel-IPMC in einer sternförmigen Umgebung nicht um ein IPMB-Hub-Gerät handelt, wird das Argument *link* nicht verwendet. Es werden Informationen zum Status von IPMB-A und IPMB-B auf dem IPM-Zielcontroller angezeigt.

Wenn es sich in der sternförmigen Umgebung beim IPM-Zielcontroller um ein IPMB-Hub-Gerät handelt, wird der Befehl wie folgt angewendet:

- Wenn *link* fehlt, zeigt der Befehl Informationen zum Status aller sternförmigen IPMB-Links an. Der Status wird anhand der Sensordaten bestimmt, die für die IPMB-Linksensoren des IPM-Controllers vorliegen.
- Wenn *link* angegeben ist, zeigt der Befehl Informationen zum entsprechenden sternförmigen IPMB-Link (1 bis 95) an. Der Status des Links wird anhand des Status ermittelt, der für den IPMB-Linksensor des IPM-Controllers vorliegt.

In beiden Fällen werden Informationen zum Status von IPMB-A und IPMB-B angezeigt.

### Beispiele:

Anzeigen des aktuellen Status von IPMB-0 auf dem IPMC unter der IPMB-Adresse 92h.

```
# clia getipmbstate 92
IPM Sentry Shelf Manager Command Line Interpreter

92: LUN: 0, Sensor # 1 ("IPMB LINK")
 Bus Status: 0x8  (IPMB-A Enabled, IPMB-B Enabled)
 IPMB A State: 0x8  (LocalControl, No failure)
 IPMB B State: 0x8  (LocalControl, No failure)
```

Anzeigen des aktuellen Status von Link 8 für den Shelf Manager in der sternförmigen Umgebung.

```
# clia getipmbstate 20 8
IPM Sentry Shelf Manager Command Line Interpreter

20: Link: 8, LUN: 0, Sensor # 12 ("IPMB LINK 8")
 Bus Status: 0x8  (IPMB-A Enabled, IPMB-B Enabled)
 IPMB A State: 0x8  (LocalControl, No failure)
 IPMB B State: 0x8  (LocalControl, No failure)
```

## getlanconfig

### Syntax:

`getlanconfig` *kanal* [*parametername* [*weitere_parameter*]]
`getlanconfig` *kanal* [*parameternummer* [*weitere_parameter*]]

## Zweck:

Dieser Befehl zeigt den Wert des angegebenen LAN-Konfigurationsparameters auf dem angegebenen Kanal an. Wenn Sie keinen Konfigurationsparameternamen bzw. eine -nummer angeben, werden alle Konfigurationsparameter für den angegebenen Kanal angezeigt.

In TABELLE A-1 sind Namen und Nummern von LAN-Konfigurationsparametern aufgeführt, die vom Befehl `getlanconfig` unterstützt werden:

**TABELLE A-1** LAN-Konfigurationsparameter für `getlanconfig`

| Parametername | Nummer | Beschreibung |
|---|---|---|
| `auth_support` | 1 | Ein 8-Bit-Wert, der Authentifizierungstyp-Supportflags für den LAN-Kanal enthält. |
| `auth_enables` | 2 | Fünf 8-Bit-Werte, die Authentifizierungstypen-Aktivierungsflags für Rückmeldungs-, Benutzer-, Bediener-, Administrator- und OEM-Berechtigungsebenen für den LAN-Kanal enthalten. |
| `ip` | 3 | Ein Zeichenfolgenwert, der die IP-Adresse enthält, die dem LAN-Kanal im Dezimalnotationformat mit Punkten zugewiesen ist (z. B. `192.168.0.15`). |
| `ip_source` | 4 | Ein Wert, der die Quelle der zugewiesenen IP-Adresse codiert. |
| `mac` | 5 | Ein Zeichenfolgenwert, der die MAC-Adresse enthält, die dem LAN-Kanal in Form von sechs hexadezimalen Byte-Werten mit dem Symbol `:` als Trennzeichen zugewiesen ist (z. B. `00:A0:24:C6:18:2F`). |
| `subnet_mask` | 6 | Ein Zeichenfolgenwert, der die Teilnetzmaske enthält, die dem LAN-Kanal im Dezimalnotationformat mit Punkten zugewiesen ist (z. B. `255.255.255.0`). |
| `ipv4_hdr_param` | 7 | Drei 8-Bit-Werte, die verschiedene IPv4-Headerparameter zum Senden von RMCP-Paketen enthalten:<br>• Lebensdauer<br>• IP-Headerflags (Bits [7:5])<br>• Vorrangigkeit (Bits [7;5]) und Diensttyp (Bits [4:1]) |
| `pri_rmcp_port` | 8 | Ein 16-Bit-Wert, der die primäre RMCP-Portnummer enthält (verwendeter Port für die normale RMCP-Kommunikation). |
| `sec_rmcp_port` | 9 | Ein 16-Bit-Wert, der die sekundäre RMCP-Portnummer enthält (verwendeter Port für die sichere RMCP-Kommunikation). |

**TABELLE A-1** LAN-Konfigurationsparameter für `getlanconfig` *(Fortsetzung)*

| | | |
|---|---|---|
| `arp_control` | 10 | Zwei Flags, die das Verhalten des Adressabbildungs-protokolls (Address Resolution Protocol, ARP) auf dem LAN-Kanal steuern:<br>• Aktivieren der Antworten auf ARP-Anforderungen<br>• Aktivieren des Sendens von nicht angeforderten ARPs |
| `arp_interval` | 11 | Das Intervall für nicht angeforderte ARPs in Sekunden und Festpunktformat (ggf. mit einem Teil als Bruch). |
| `dft_gw_ip` | 12 | Ein Zeichenfolgenwert, der die IP-Adresse des Standardgateways in Dezimalnotation mit Punkten enthält. |
| `dft_gw_mac` | 13 | Ein Zeichenfolgenwert, der die MAC-Adresse des Standardgateways in Form von sechs hexadezimalen Byte-Werten mit Doppelpunkten als Trennzeichen enthält (`:`). |
| `backup_gw_ip` | 14 | Ein Zeichenfolgenwert, der die IP-Adresse des Sicherungsgateways in Dezimalnotation mit Punkten enthält. |
| `backup_gw_mac` | 15 | Ein Zeichenfolgenwert, der die MAC-Adresse des Sicherungsgateways in Form von sechs hexadezimalen Byte-Werten mit Doppelpunkten als Trennzeichen enthält (`:`). |
| `community` | 16 | Ein Zeichenfolgenwert (bis zu 18 Symbole), der bei PET-Unterbrechungen in das Feld `Community String` eingefügt wird. |
| `destination_count` | 17 | Die maximale Anzahl an LAN-Warnmeldungszielen, die für den LAN-Kanal unterstützt werden. |
| `destination_type` | 18 | Der Zieltyp, der mithilfe der angegebenen Auswahlwerkzeugfestlegung (Set Selector) identifiziert wird. Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle Zieltypen angezeigt. Jeder Zieltypeintrag enthält die folgenden Felder:<br>• Zieltyp (0 bis 7)<br>• Flag für Warnmeldungsbestätigung<br>• Zeitlimit für die Warnmeldungsbestätigung/Wiederholungsintervall in Sekunden (1 bis 256)<br>• Anzahl der Wiederholungsversuche (0 bis 7) |

**TABELLE A-1** LAN-Konfigurationsparameter für `getlanconfig` *(Fortsetzung)*

| | | |
|---|---|---|
| `destination_address` | 19 | Die Zieladressen, die der angegebenen Auswahlwerkzeugfestlegung zugeordnet sind. Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle Zieladressen angezeigt. Jede Zieladresse enthält die folgenden Felder:<br>• Auswahlwerkzeug für Gateway: 0 – Standardeinstellung verwenden, 1 – Sicherung verwenden<br>• IP-Adresse (Zeichenfolge im Dezimalformat mit Punkten)<br>• MAC-Adresse (Zeichenfolge mit sechs hexadezimalen Byte-Werten mit Doppelpunkten als Trennzeichen [ : ]) |

### Beispiel:

Die folgenden Unterabschnitte enthalten ausführlichere Informationen zu den einzelnen unterstützten Parametern.

Abrufen und Anzeigen der LAN-Parametertabelle für Kanal 1.

```
# clia getlanconfig 1

Authentication Type Support: 0x15 (None MD5 Straight Password/Key)
Authentication Type Enables: 0x00
    User level: 0x15 ( None MD5 Straight Password/Key )
    Operator level: 0x15 ( None MD5 Straight Password/Key )
    Administrator level: 0x15 ( None MD5 Straight Password/Key )
    OEM level: 0x00
IP Address: 172.16.2.203
IP Address Source: Static Address (Manually Configured) (01)
MAC Address: 90:91:91:91:91:91
Subnet Mask: 255.255.255.0
IPv4 Header Parameters: 0x40:0x40:0x10
Primary RMCP Port Number: 0x026f
Secondary RMCP Port Number: 0x0298
BMC-generated ARP Control: 02
    Enable BMC-generated Gratuitous Response
Gratuitous ARP Interval: 2.0 seconds
Default Gateway Address: 0.0.0.0
Default Gateway MAC Address: N/A
Backup Gateway Address: 0.0.0.0
Backup MAC Address: N/A
```

```
Community String: "public"
Number of Destinations: 16
#
```

# auth_support

### Syntax:

getlanconfig *kanal* auth_support
getlanconfig *kanal* 1

### Zweck:

Dieser Befehl zeigt den aktuellen Wert des LAN-Parameters `auth_support` an. Dieser Parameter gibt an, welche Authentifizierungstypen vom Shelf Manager unterstützt werden. Es wird ein einzelnes Byte verwendet, das als Bitmaske behandelt wird und bei dem die Bits wie folgt definiert sind:

- 0x01 – Kein
- 0x02 – MD2
- 0x04 – MD5
- 0x10 – Straight-Passwort/-Schlüssel
- 0x20 – OEM-abhängig

Andere Bits sind reserviert und müssen auf 0 gesetzt sein.

Zusätzlich zum hexadezimalen Rohwert werden auch symbolische Werte angezeigt, die für die Bits festgelegt sind.

### Beispiel:

```
# clia getlanconfig 1 auth_support
IPM Sentry Shelf Manager Command Line Interpreter
Authentication Type Support: 0x15 ( None MD5 Straight Password/Key
#
```

# auth_enables

### Syntax:

getlanconfig *kanal* auth_enables
getlanconfig *kanal* 2

### Zweck:

Dieser Befehl zeigt den aktuellen Wert des LAN-Parameters auth_enables an. Dieser Parameter gibt an, welche Authentifizierungstypen vom Shelf Manager für die fünf unterstützten Berechtigungsebenen momentan aktiviert sind (Rückmeldung, Benutzer, Bediener, Administrator und OEM). Die Darstellung erfolgt über fünf Bytes, die jeweils für die entsprechende Berechtigungsebene stehen und als Bitmaske behandelt werden. Die Bits sind dabei wie folgt definiert:

- 0x01 – Kein
- 0x02 – MD2
- 0x04 – MD5
- 0x10 – Straight-Passwort/-Schlüssel
- 0x20 – OEM-abhängig

Andere Bits sind reserviert und müssen auf 0 gesetzt sein.

Zusätzlich zu den hexadezimalen Rohwerten werden auch symbolische Werte angezeigt, die für die Bits festgelegt sind.

### Beispiel:

```
# clia getlanconfig 1 auth_enables
IPM Sentry Shelf Manager Command Line Interpreter
Authentication Type Enables:
    Callback level: 0x00
    User level: 0x15 ( None MD5 Straight Password/Key )
    Operator level: 0x15 ( None MD5 Straight Password/Key )
    Administrator level: 0x15 ( None MD5 Straight Password/Key )
    OEM level: 0x00
```

# ip

### Syntax:

getlanconfig *kanal* ip
getlanconfig *kanal* 3

### Zweck:

Dieser Befehl zeigt die aktuelle IP-Adresse, die vom Kanal verwendet wird, in Dezimalnotation mit Punkten an.

### Beispiel:

```
# clia getlanconfig 1 ip
IPM Sentry Shelf Manager Command Line Interpreter
IP Address: 172.16.2.203
```

# ip_source

### Syntax:

getlanconfig *kanal* ip_source
getlanconfig *kanal* 4

### Zweck:

Dieser Befehl zeigt den aktuellen Wert des LAN-Parameters ip_source an. Dieser Parameter gibt die Quelle der IP-Adresse an, die vom Shelf Manager verwendet wird. Er wird durch ein einzelnes Byte angegeben, das einen der folgenden Werte haben kann:

- 0 – Kein Angabe
- 1 – Statische Adresse (manuell konfiguriert)
- 2 – Adresse wird vom Shelf Manager durch Ausführung des DHCP ermittelt
- 3 – Adresse wird vom BIOS oder von der Systemsoftware geladen
- 4 – Adresse wird vom Shelf Manager durch Ausführung eines anderen Adresszuweisungsprotokolls ermittelt

Andere Werte sind reserviert.

Zusätzlich zum hexadezimalen Rohwert wird der symbolische Wert angezeigt.

**Beispiel:**

```
# clia getlanconfig 1 ip_source
IPM Sentry Shelf Manager Command Line Interpreter
IP Address Source: Static Address (Manually Configured) (0x01)
#
```

# mac

**Syntax:**

getlanconfig *kanal* mac
getlanconfig *kanal* 5

**Zweck:**

Dieser Befehl zeigt die aktuelle MAC-Adresse, die vom Kanal verwendet wird, in Form von sechs hexadezimalen Bytes mit Doppelpunkten als Trennzeichen an.

**Beispiel:**

```
# clia getlanconfig 1 mac
IPM Sentry Shelf Manager Command Line Interpreter
MAC Address: 90:91:91:91:91:91
#
```

# subnet_mask

**Syntax:**

getlanconfig *kanal* subnet_mask
getlanconfig *kanal* 6

**Zweck:**

Dieser Befehl zeigt die aktuelle IP-Teilnetzmaske, die vom Kanal verwendet wird, in Dezimalnotation mit Punkten an.

**Beispiel:**

```
# clia getlanconfig 1 subnet_mask
IPM Sentry Shelf Manager Command Line Interpreter
Subnet Mask: 255.255.255.0
#
```

# ipv4_hdr_param

**Syntax:**

getlanconfig *kanal* ipv4_hdr_param
getlanconfig *kanal* 7

**Zweck:**

Dieser Befehl zeigt die aktuellen IP 4-Headerparameter an. Diese werden als drei Einzelbyte-Werte in hexadezimaler Notation mit Doppelpunkten als Trennzeichen angegeben. Der Inhalt der Bytes entspricht Abschnitt 19.2 der Spezifikation IPMI 1.5.

**Beispiel:**

```
# clia getlanconfig 1 ipv4_hdr_param
IPM Sentry Shelf Manager Command Line Interpreter
IPv4 Header Parameters: 0x40:0x40:0x10
#
```

# pri_rmcp_port

**Syntax:**

getlanconfig *kanal* pri_rmcp_port
getlanconfig *kanal* 8

**Zweck:**

Dieser Befehl zeigt den aktuellen primären RMCP-Port, der vom Kanal verwendet wird, in hexadezimaler Notation an. Dies ist der Port, der für normale RMCP-Interaktionen verwendet wird.

**Beispiel:**

```
# clia getlanconfig 1 pri_rmcp_port
IPM Sentry Shelf Manager Command Line Interpreter
Primary RMCP Port Number: 0x026f
#
```

# sec_rmcp_port

**Syntax:**

`getlanconfig` *kanal* `sec_rmcp_port`
`getlanconfig` *kanal* `9`

**Zweck:**

Dieser Befehl zeigt den aktuellen sekundären RMCP-Port, der vom Kanal verwendet wird, in hexadezimaler Notation an. Dies ist der Port, der für sichere RMCP-Interaktionen verwendet wird.

**Beispiel:**

```
# clia getlanconfig 1 sec_rmcp_port
IPM Sentry Shelf Manager Command Line Interpreter
Primary RMCP Port Number: 0x0298
#
```

# arp_control

**Syntax:**

`getlanconfig` *kanal* `arp_control`
`getlanconfig` *kanal* `10`

**Zweck:**

Dieser Befehl zeigt den aktuellen Wert des LAN-Parameters `arp_control` an. Dieser Parameter gibt die zusätzliche ARP-Unterstützung an, die vom Shelf Manager bereitgestellt wird. Es wird ein einzelnes Byte verwendet, das als Bitmaske behandelt wird und bei dem die Bits wie folgt definiert sind:

- 1 – Vom Shelf Manager erzeugte nicht angeforderte ARPs aktivieren
- 2 – Vom Shelf Manager erzeugte ARP-Antworten aktivieren

Andere Bits sind reserviert und müssen auf 0 gesetzt sein.

Zusätzlich zum hexadezimalen Rohwert werden auch symbolische Werte angezeigt, die für die Bits festgelegt sind.

**Beispiel:**

```
# clia getlanconfig 1 arp_control
IPM Sentry Shelf Manager Command Line Interpreter
BMC-generated ARP Control: 02
    Enable BMC-generated Gratuitous Response
#
```

# arp_interval

**Syntax:**

`getlanconfig` *kanal* `arp_interval`
`getlanconfig` *kanal* `11`

**Zweck:**

Dieser Befehl zeigt das aktuelle ARP-Intervall an, das vom Kanal verwendet wird. Der Wert wird als Anzahl an Sekunden im numerischen Festpunktformat angezeigt.

**Beispiel:**

```
# clia getlanconfig 1 arp_interval
IPM Sentry Shelf Manager Command Line Interpreter
Gratuitous ARP Interval: 2.0 seconds
#
```

# dft_gw_ip

**Syntax:**

`getlanconfig` *kanal* dft_gw_ip
`getlanconfig` *kanal* 12

**Zweck:**

Dieser Befehl zeigt die IP-Adresse des Standardgateways, die vom Kanal verwendet wird, in Dezimalnotation mit Punkten an.

**Beispiel:**

```
# clia getlanconfig 1 dft_gw_ip
IPM Sentry Shelf Manager Command Line Interpreter
Default Gateway Address: 0.0.0.0
#
```

## dft_gw_mac

**Syntax:**

getlanconfig *kanal* dft_gw_mac
getlanconfig *kanal* 13

**Zweck:**

Dieser Befehl zeigt die MAC-Adresse des Standardgateways, die vom Kanal verwendet wird, in Form von sechs hexadezimalen Bytes mit Doppelpunkten als Trennzeichen an.

**Beispiel:**

```
# clia getlanconfig 1 dft_gw_mac
IPM Sentry Shelf Manager Command Line Interpreter
Default Gateway MAC Address: N/A
#
```

## backup_gw_ip

**Syntax:**

getlanconfig *kanal* backup_gw_ip
getlanconfig *kanal* 14

**Zweck:**

Dieser Befehl zeigt die IP-Adresse des Sicherungsgateways, die vom Kanal verwendet wird, in Dezimalnotation mit Punkten an.

### Beispiel:

```
# clia getlanconfig 1 backup_gw_ip
IPM Sentry Shelf Manager Command Line Interpreter
Backup Gateway Address: 0.0.0.0
#
```

## backup_gw_mac

### Syntax:

getlanconfig *kanal* backup_gw_mac
getlanconfig *kanal* 15

### Zweck:

Dieser Befehl zeigt die MAC-Adresse des Sicherungsgateways, die vom Kanal verwendet wird, in Form von sechs hexadezimalen Bytes mit Doppelpunkten als Trennzeichen an.

### Beispiel:

```
# clia getlanconfig 1 backup_gw_mac
IPM Sentry Shelf Manager Command Line Interpreter
Backup Gateway MAC Address: N/A
#
```

## community

### Syntax:

getlanconfig *kanal* community
getlanconfig *kanal* 16

### Zweck:

Dieser Befehl zeigt den Community-Zeichenfolgenparameter an, der in PET-Unterbrechungen verwendet wird.

**Beispiel:**

```
# clia getlanconfig 1 community
IPM Sentry Shelf Manager Command Line Interpreter
Community String: "public"
#
```

# destination_count

**Syntax:**

getlanconfig *kanal* destination_count
getlanconfig *kanal* 17

**Zweck:**

Dieser Befehl zeigt die maximale Anzahl an Warnmeldungszielen an, die für den Kanal verfügbar sind. Es handelt sich um einen Konfigurationsparameter für den IPM Sentry Shelf Manager, der nur über die Konfigurationsdatei `shelfman` geändert werden kann.

**Beispiel:**

```
# clia getlanconfig 1 destination_count
IPM Sentry Shelf Manager Command Line Interpreter
Number of Destinations: 16
#
```

# destination_type

**Syntax:**

getlanconfig *kanal* destination_type [*auswahlwerkzeug_festlegen*]
getlanconfig *kanal* 18 [*auswahlwerkzeug_festlegen*]

**Zweck:**

Dieser Befehl zeigt das Element der Zieltabelle an, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 0-basiert. Der Wert 0 wird für die Adressierung des flüchtigen Ziels verwendet. Für das Ziel werden die folgenden Informationen angezeigt:

- Zielauswahlwerkzeug

- Typ des Warnmeldungsziels (PET-Unterbrechung oder OEM-Ziel; Angabe, ob die Warnmeldung bestätigt werden soll)
- Zeitlimit für Warnmeldungsbestätigung
- Anzahl an Wiederholungsversuchen

Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle aktiven Ziele mit ihren Nummern angezeigt.

### Beispiele:

```
# clia getlanconfig 1 destination_type 2
IPM Sentry Shelf Manager Command Line Interpreter
DST Type # 2, Type: Acknowledged PET Trap Destination (0x80), ACK
Timeout / Retry Interval: 3 seconds, Retries: 5

# clia getlanconfig 1 destination_type
IPM Sentry Shelf Manager Command Line Interpreter
DST Type # 0, Type: Acknowledged reserved (0x81), ACK Timeout /
Retry Interval: 2 seconds, Retries: 6
DST Type # 1, Type: Unacknowledged reserved (0x02), ACK Timeout /
Retry Interval: 3 seconds, Retries: 4
DST Type # 2, Type: Acknowledged PET Trap Destination (0x80), ACK
Timeout / Retry Interval: 3 seconds, Retries: 5
#
```

## destination_address

### Syntax:

`getlanconfig` *kanal* `destination_address [`*auswahlwerkzeug_festlegen*`]`
`getlanconfig` *kanal* `19 [`*auswahlwerkzeug_festlegen*`]`

### Zweck:

Dieser Befehl zeigt das Element der Zieladresstabelle an, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 0-basiert. Der Wert 0 wird für die Adressierung des flüchtigen Ziels verwendet. Für das Ziel werden die folgenden Informationen angezeigt:

- Zielauswahlwerkzeug
- Adressenformat (standardmäßig IP und MAC)
- IP-Zieladresse
- MAC-Zieladresse
- Zu verwendendes Gateway (Standard oder Sicherung).

Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle aktiven Zieladressen mit ihren Nummern angezeigt.

### Beispiel:

```
# clia getlanconfig 1 destination_address 2
IPM Sentry Shelf Manager Command Line Interpreter
DST Addresses # 2, Address Format: IPv4 IP Address followed by DIX
ethernet / 802.3 MAC Address (0x00)
    Gateway: Default (0x00), Alerting IP: 172.16.2.100, Alerting
MAC: 90:93:93:93:93:93
#
```

## getpefconfig

### Syntax:

getpefconfig
getpefconfig *parametername* [*weitere_parameter*]
getpefconfig *parameternummer* [*weitere_parameter*]

### Zweck:

Dieser Befehl zeigt den Wert des angegebenen PEF-Konfigurationsparameters an. Wenn Sie keinen Konfigurationsparameternamen und keine Konfigurationsparameternummer angeben, werden alle PEF-Konfigurationsparameter angezeigt.

In TABELLE A-2 sind die Namen und Nummern der PEF-Konfigurationsparameter aufgeführt:

**TABELLE A-2** PEF-Konfigurationsparameter

| Parametername | Nummer | Beschreibung |
|---|---|---|
| control | 1 | Ein 8-Bit-Wert, der für PEF-Steuerflags steht (PEF aktivieren, PEF-Startverzögerung aktivieren usw.). |
| action_control | 2 | Ein 8-Bit-Wert, der für globale Steuerflags für PEF-Aktionen steht (Zurücksetzen aktivieren, Herunterfahren aktivieren usw.). |
| startup_delay | 3 | Zeitraum für die PEF-Verzögerung nach dem Hochfahren bzw. Zurücksetzen des Systems in Sekunden. |

**TABELLE A-2** PEF-Konfigurationsparameter *(Fortsetzung)*

| | | |
|---|---|---|
| alert_startup_delay | 4 | Zeitraum für die Verzögerungswarnmeldungen nach dem Hochfahren bzw. Zurücksetzen des Systems in Sekunden. |
| event_filter_count | 5 | Maximale Anzahl an Ereignisfiltern. |
| event_filter | 6 | Ein Ereignisfilter-Tabelleneintrag, für den die angegebene Auswahlwerkzeugfestlegung steht. Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle aktiven Ereignisfilter angezeigt. |
| event_filter_data1 | 7 | Das erste Byte des Ereignisfilter-Tabelleneintrags, für den die angegebene Auswahlwerkzeugfestlegung steht. Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle aktiven Ereignisfilter angezeigt. |
| alert_policy_count | 8 | Maximale Anzahl an Warnverfahren. |
| alert_policy | 9 | Ein Warnverfahren-Tabelleneintrag, für den die angegebene Auswahlwerkzeugfestlegung steht. Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle aktiven Warnverfahren angezeigt. |
| system_guid | 10 | Eine GUID, die bei der PET-Unterbrechung in das GUID-Feld eingefügt wird. |
| alert_string_count | 11 | Maximale Anzahl an Warnmeldungszeichenfolgen. |
| alert_string_key | 12 | Ein Warnmeldungszeichenfolgen-Schlüssel, für den die angegebene Auswahlwerkzeugfestlegung steht. Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle Warnmeldungszeichenfolgen-Schlüssel angezeigt. |
| alert_string | 13 | Eine Warnmeldungszeichenfolge, für die die angegebene Auswahlwerkzeugfestlegung steht. Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle Warnmeldungszeichenfolgen angezeigt. |
| oem_filter_count | 96 | Maximale Anzahl an OEM-Filtern. |
| oem_filter | 97 | Ein OEM-Filter-Tabelleneintrag, für den die angegebene Auswahlwerkzeugfestlegung steht. Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle aktiven Ereignisfilter angezeigt. |

### Beispiel:

Die folgenden Unterabschnitte enthalten ausführlichere Informationen zu den einzelnen unterstützten Parametern.

Abrufen und Anzeigen der gesamten PEF-Parametertabelle.

```
# clia getpefconfig
IPM Sentry Shelf Manager Command Line Interpreter
PEF parameters:
    PEF control: 0x00
    PEF Action Global Control: 0x00
    PEF Startup Delay: 60 seconds
    PEF Alert Startup Delay: 60 seconds
    PEF Number of Event Filters: 64
    PEF Number of OEM Filters: 16
    Active Event Filters:
        None
    Active event filter data:
        None
    Alert Policies Count: 64
    Policy:
        None
    PEF GUID: Using the system GUID
Alert Strings Count: 64
    Alert string key:
        None
    Alert Strings:
        None
#
```

# control

### Syntax:

```
getpefconfig control
getpefconfig 1
```

### Zweck:

Dieser Befehl zeigt den aktuellen Wert des PEF-Parameters `control` an. Für den Parameter wird ein einzelnes Byte verwendet, das als Bitmaske behandelt wird und bei dem die Bits wie folgt definiert sind:

- 0x01 – PEF aktivieren
- 0x02 – Erzeugen von Ereignismeldungen für PEF-Aktionen aktivieren
- 0x04 – PEF-Startverzögerungen beim Hochfahren und Zurücksetzen des Systems aktivieren
- 0x08 – Startverzögerungen für PEF-Warnmeldungen aktivieren

Andere Bits sind reserviert und müssen auf 0 gesetzt sein.

### Beispiel:

```
# clia getpefconfig control
IPM Sentry Shelf Manager Command Line Interpreter
PEF control: 0x07
        Enable PEF
        Enable Event Message for PEF Actions
        Enable PEF Startup Delay
#
```

## action_control

### Syntax:

```
getpefconfig action_control
getpefconfig 2
```

### Zweck:

Dieser Befehl zeigt den aktuellen Wert des PEF-Parameters `action_control` an. Für den Parameter wird ein einzelnes Byte verwendet, das als Bitmaske behandelt wird und bei dem die Bits wie folgt definiert sind:

- 0x01 – Warnmeldungsaktion aktivieren
- 0x02 – Aktion zum Herunterfahren aktivieren
- 0x04 – Aktion zum Zurücksetzen aktivieren
- 0x08 – Aktion für Neustart (Power Cycle) aktivieren
- 0x10 – OEM-Aktion aktivieren
- 0x20 – Diagnoseunterbrechung aktivieren

Andere Bits sind reserviert und müssen auf 0 gesetzt sein.

### Beispiel:

```
# clia getpefconfig action_control
IPM Sentry Shelf Manager Command Line Interpreter
PEF Action Global Control: 0x3f
        Enable Alert Action
        Enable Power Down Action
        Enable Reset Action
        Enable Power Cycle Action
        Enable OEM Action
        Enable Diagnostic Interrupt
#
```

## startup_delay

### Syntax:

```
getpefconfig startup_delay
getpefconfig 3
```

### Zweck:

Dieser Befehl zeigt den aktuellen Wert des PEF-Parameters `startup_delay` an. Bei diesem Parameter handelt es sich um ein einzelnes Byte, das für die Anzahl an Sekunden steht, um die das PEF-Dienstprogramm beim Starten verzögert wird.

### Beispiel:

```
# clia getpefconfig startup_delay
IPM Sentry Shelf Manager Command Line Interpreter
    PEF Startup Delay: 60 seconds
#
```

## alert_startup_delay

### Syntax:

```
getpefconfig startup_delay
getpefconfig 4
```

### Zweck:

Dieser Befehl zeigt den aktuellen Wert des PEF-Parameters `alert_startup_delay` an. Bei diesem Parameter handelt es sich um ein einzelnes Byte, das für die Anzahl an Sekunden steht, um die das Warnmeldungsdienstprogramm beim Starten verzögert wird.

### Beispiel:

```
# clia getpefconfig alert_startup_delay
IPM Sentry Shelf Manager Command Line Interpreter
PEF Alert Startup Delay: 60 seconds
#
```

## event_filter_count

### Syntax:

```
getpefconfig event_filter_count
getpefconfig 5
```

### Zweck:

Dieser Befehl zeigt den aktuellen Wert des PEF-Parameters `event_filter_count` an. Dieser schreibgeschützte Wert gibt die Größe der Ereignisfiltertabelle an. Bei diesem Wert handelt sich um einen Konfigurationsparameter für den IPM Sentry Shelf Manager, der nur über die Konfigurationsdatei `shelfman` geändert werden kann.

### Beispiel:

```
# clia getpefconfig event_filter_count
IPM Sentry Shelf Manager Command Line Interpreter
PEF Number of Event Filters: 64
#
```

## event_filter

### Syntax:

`getpefconfig event_filter [`*auswahlwerkzeug_festlegen*`]`
`getpefconfig 6 [`*auswahlwerkzeug_festlegen*`]`

### Zweck:

Dieser Befehl zeigt das Element der Ereignisfiltertabelle an, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 1-basiert. Für jeden Ereignisfilter werden die folgenden Informationen angezeigt:

- Filterkonfiguration: Hinweis, ob der Filter durch die Software konfiguriert oder vom Hersteller vorkonfiguriert wurde
- Ereignisfilter-Aktionsmaske
- Nummer des Warnverfahrens
- Ereignisschweregrad
- Ereignisquellenadresse für Abgleich (255 = beliebige Adresse)
- Quellkanal/LU-Nummer für Abgleich (255 = Quellkanal/LU-Nummer beliebig)
- Sensortyp für Abgleich
- Sensornummer für Abgleich
- Ereignisauslöser (Ereignis-/Lesetyp) für Abgleich
- Ereignisversatzmaske
- Masken AND, Compare 1 und Compare 2 für Ereignisdatenbytes 1, 2 und 3.

Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle aktiven Ereignisfilter-Tabelleneinträge mit ihren Nummern angezeigt.

### Beispiel:

```
# clia getpefconfig event_filter 2

IPM Sentry Shelf Manager Command Line Interpreter
    Active Event Filters:
    0x02: Software Configurable Filter
        Action Mask: 0x01
        Policy Number: 1, Severity: Critical Condition
        Source Address: 0x20, LUN: 3, Channel: 15
        Sensor Type: Hot Swap (0xf0), Sensor # 255 (ANY)
        Event Trigger: 0xff (ANY), Event Offset Mask: 0xffff
        0: AND: 0x0f, CMP1: 0xff, CMP2: 0x00
        1: AND: 0x00, CMP1: 0x00, CMP2: 0x00
        2: AND: 0xff, CMP1: 0xff, CMP2: 0x00
#
```

## event_filter_data1

### Syntax:

getpefconfig event_filter_data1 [*auswahlwerkzeug_festlegen*]
getpefconfig 7 [*auswahlwerkzeug_festlegen*]

### Zweck:

Dieser Befehl zeigt das erste Byte des Elements der Ereignisfiltertabelle an, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 1-basiert. Dieses Byte wird im hexadezimalen Format angezeigt. Die Bits haben dabei die folgende Bedeutung:

- 0x80 – Dieser Filter ist aktiviert.
- 0x40 – Dieser Filter ist vom Hersteller vorkonfiguriert und darf mithilfe der Software nicht geändert werden.

Andere Bits sind reserviert und müssen auf 0 gesetzt sein.

Wenn keine Auswahlwerkzeugfestlegung angegeben ist, wird für die einzelnen aktiven Ereignisfilter-Tabelleneinträge jeweils das erste Byte mit den entsprechenden Filternummern angezeigt.

### Beispiel:

```
# clia getpefconfig event_filter_data1 2
IPM Sentry Shelf Manager Command Line Interpreter

    Active event filter data:
        0x02: 0x80 Enabled 1, Configuration: 0 ("Software
Configurable Filter")
#
```

# alert_policy_count

### Syntax:

```
getpefconfig alert_policy_count
getpefconfig 8
```

### Zweck:

Dieser Befehl zeigt den aktuellen Wert des PEF-Parameters `alert_policy_count` an. Dieser schreibgeschützte Wert gibt die Größe der Warnverfahrentabelle an. Bei diesem Wert handelt sich um einen Konfigurationsparameter für den IPM Sentry Shelf Manager, der nur über die Konfigurationsdatei `shelfman` geändert werden kann.

### Beispiel:

```
# clia getpefconfig alert_policy_count
IPM Sentry Shelf Manager Command Line Interpreter
    Alert Policies Count: 64
#
```

# alert_policy

### Syntax:

getpefconfig alert_policy [*auswahlwerkzeug_festlegen*]
getpefconfig 9 [*auswahlwerkzeug_festlegen*]

### Zweck:

Dieser Befehl zeigt das Element der Warnverfahrentabelle an, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 1-basiert. Für jedes Warnverfahren werden die folgenden Informationen angezeigt:

- Nummer des Verfahrens
- Typ des Verfahrens (in Bezug auf die Warnmeldung, die an das vorherige Ziel gesendet wurde)
- Zielkanalnummer
- Zielauswahlwerkzeug
- Warnmeldungszeichenfolgen-Schlüssel

Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle aktiven Warnverfahren-Tabelleneinträge mit ihren Nummern angezeigt.

**Beispiel:**

```
# clia getpefconfig alert_policy 2
IPM Sentry Shelf Manager Command Line Interpreter

    Policy:
       0x02: Policy# 5, Policy Type: 0, Channel: 1, DST: 1, Alert
String Sel: 1
#
```

# system_guid

**Syntax:**

```
getpefconfig system_guid
getpefconfig 10
```

**Zweck:**

Dieser Befehl zeigt den aktuellen Wert des PEF-Parameters `system_guid` an. Dieser Parameter steht für die GUID, die in einer PET-Unterbrechungs-PDU an ein Warnmeldungsziel gesendet wird. Diese GUID kann als separate GUID oder identisch mit der System-GUID definiert werden (die Sie über den Befehl `Get System GUID IPMI` abrufen können).

**Beispiel:**

```
# clia getpefconfig system_guid
IPM Sentry Shelf Manager Command Line Interpreter
    PEF GUID: 23662f7f-ba1b-4b65-8808-94ca09c9bbb0
#
```

# alert_string_count

**Syntax:**

```
getpefconfig alert_string_count
getpefconfig 11
```

### Zweck:

Dieser Befehl zeigt den aktuellen Wert des PEF-Parameters `alert_string_count` an. Dieser schreibgeschützte Wert gibt die Größe der Warnmeldungszeichenfolgen-Schlüsseltabelle an. Hierbei handelt es sich um die maximale Anzahl an Warnmeldungszeichenfolgen, die gleichzeitig verwendet werden. Bei diesem Wert handelt sich um den Konfigurationsparameter für den IPM Sentry Shelf Manager, der nur über die Konfigurationsdatei `shelfman` geändert werden kann.

### Beispiel:

```
# clia getpefconfig alert_string_count

IPM Sentry Shelf Manager Command Line Interpreter

    Alert Strings Count: 64
#
```

## alert_string_key

### Syntax:

`getpefconfig alert_string_key [`*auswahlwerkzeug_festlegen*`]`
`getpefconfig 12 [`*auswahlwerkzeug_festlegen*`]`

### Zweck:

Dieser Befehl zeigt das Element der Warnmeldungszeichenfolgen-Schlüsseltabelle an, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 1-basiert. Sie können einen Indexwert von 0 verwenden, um die flüchtige Warnmeldungszeichenfolge anzugeben. Jeder Schlüssel weist einem Ereignisfilter eine Warnmeldungszeichenfolge zu, damit Warnmeldungen erzeugt werden können. Für jede Warnmeldungszeichenfolge werden die folgenden Informationen angezeigt:

- Schlüsselnummer der Warnmeldungszeichenfolge
- Zugeordnete Ereignisfilternummer
- Zugeordnete Nummer der Warnmeldungszeichenfolge

Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden die Tabelleneinträge aller aktiven Warnmeldungszeichenfolgen-Schlüssel mit ihren Nummern angezeigt.

**Beispiel:**

```
# clia getpefconfig alert_string_key 2
IPM Sentry Shelf Manager Command Line Interpreter
   Alert string key: set selector 2, event_filter 0x10, string_set
0x11
#
```

# alert_string

**Syntax:**

getpefconfig alert_string [*auswahlwerkzeug_festlegen*]
getpefconfig 13 [*auswahlwerkzeug_festlegen*]

**Zweck:**

Dieser Befehl zeigt das Element der Warnmeldungszeichenfolgen-Schlüsseltabelle an, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 1-basiert. Sie können einen Indexwert von 0 verwenden, um die flüchtige Warnmeldungszeichenfolge anzugeben. Dieser Befehl zeigt die gesamte Zeichenfolge auf einmal an.

Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle definierten Warnmeldungszeichenfolgen mit ihren Nummern angezeigt.

**Beispiel:**

```
# clia getpefconfig alert_string 2
IPM Sentry Shelf Manager Command Line Interpreter
    Alert Strings:
         0x02: "This is the alert string"
#
```

# oem_filter_count

**Syntax:**

getpefconfig oem_filter_count
getpefconfig 96

### Zweck:

Dieser Befehl zeigt den aktuellen Wert des PEF-Parameters `oem_filter_count` an. Dieser schreibgeschützte Wert gibt die Größe der OEM-Filtertabelle an. Bei diesem Wert handelt sich um einen Konfigurationsparameter für den IPM Sentry Shelf Manager, der nur über die Konfigurationsdatei `shelfman` geändert werden kann.

Die OEM-Filtertabelle ist eine Pigeon Point Systems-definierte OEM-Erweiterung der IPMI-Spezifikation. Sie ermöglicht es, dass PEF-Parameter zusätzlich zu Plattformereignissen auf SEL-Einträge mit OEM-Zeitstempel und ohne Zeitstempel (Datensatztypbereich C0h bis FFh) angewendet werden.

### Beispiel:

```
# clia getpefconfig oem_filter_count
IPM Sentry Shelf Manager Command Line Interpreter
    PEF Number of OEM Filters: 16
#
```

## oem_filter

### Syntax:

`getpefconfig oem_filter [`*auswahlwerkzeug_festlegen*`]`
`getpefconfig 97 [`*auswahlwerkzeug_festlegen*`]`

### Zweck:

Die OEM-Filtertabelle ist eine Pigeon Point Systems-definierte OEM-Erweiterung der IPMI-Spezifikation. Sie ermöglicht es, dass PEF-Parameter zusätzlich zu Plattformereignissen auf SEL-Einträge mit OEM-Zeitstempel und ohne Zeitstempel (Datensatztypbereich C0h bis FFh) angewendet werden.

Jeder Eintrag der OEM-Filtertabelle definiert den Bereich der Datensatztypen (im Bereich der OEM-Datensatztypen), für den der jeweilige OEM-Filter gilt. Außerdem definiert er die Warnverfahrennummer, die aufgerufen wird, wenn ein Datensatz mit übereinstimmendem Datensatztyp in das SEL eingefügt wird.

Dieser Befehl zeigt das Element der OEM-Filtertabelle an, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 1-basiert. Für jeden OEM-Filter werden die folgenden Informationen angezeigt:

- Byte 1: Untergrenze für SEL-Datensatztypbereich
- Byte 2: Obergrenze für SEL-Datensatztypbereich

- Byte 3: Warnverfahrennummer, die für SEL-Einträge aufgerufen wird, deren Datensatztypen dem in Byte 1 und Byte 2 angegebenen Bereich entsprechen.

Wenn keine Auswahlwerkzeugfestlegung angegeben ist, werden alle aktiven OEM-Filter-Tabelleneinträge mit ihren Nummern angezeigt.

**Beispiel:**

```
# clia getpefconfig oem_filter
IPM Sentry Shelf Manager Command Line Interpreter
    Active OEM Filters:
    0x01: OEM range boundary 0xff:0xff, alert policy # 1
#
```

# getsensoreventenable

**Syntax:**

getsensoreventenable [*IPMB-adresse* [*sensorname* |[*lun*:]*sensornummer*]
getsensoreventenable board *n* [*sensorname* | [*lun*:]*sensornummer*]]
getsensoreventenable shm *n* [*sensorname* | [*lun*:]*sensornummer*]]

Dieser Befehl zeigt die aktuellen Ereignisaktivierungsmasken-Werte für die angegebenen Sensoren an.

Mithilfe dieses Befehls können Benutzer die Sensornummer mit der Nummer der logischen Einheit (Logical Unit Number, LUN) qualifizieren, wenn der Zielcontroller Sensoren auf mehreren LU-Nummern unterstützt. Wenn die LU-Nummer fehlt, werden Informationen zu Sensoren mit der angegebenen Sensornummer für alle LU-Nummern angezeigt. *lun* kann den Wert 0, 1 oder 3 haben. (LUN 2 ist reserviert.)

Sensornamen sind nicht durch LU-Nummern qualifiziert, da vorausgesetzt wird, dass Sensornamen innerhalb eines Controllers in der Regel eindeutig sind. Wenn in einem Controller jedoch mehrere Sensoren mit demselben Namen vorhanden sind, werden zu allen Sensoren Informationen angezeigt.

Dieser Befehl zeigt die aktuellen Sensorereignismasken-Werte für die unterstützten Ereignisse der angegebenen Sensoren an. Außerdem werden für jeden Sensor die folgenden Attribute angezeigt:

- IPMB-Adresse des als Eigentümer fungierenden IPM-Controllers
- Sensornummer, Sensorname (Geräte-ID-Zeichenfolge aus dem SDR) und die LU-Nummer, über die auf den Sensor zugegriffen werden kann
- Sensortyp

### Beispiele:

Abrufen von Ereignisaktivierungswerten für einen Temperatursensor `Local Temp` beim IPM-Controller FE.

```
# clia getsensoreventenable -v fe "Local Temp"

IPM Sentry Shelf Manager Command Line Interpreter

fe: LUN: 0, Sensor # 3 ("Local Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Assertion event mask:   0x0a80
        Assertion event for "Upper Non-Recoverable Going High"
enabled
        Assertion event for "Upper Critical Going High" enabled
       Assertion event for "Upper Non-Critical Going High" enabled
    Deassertion event mask: 0x0a80
        Deassertion event for "Upper Non-Recoverable Going High"
enabled
        Deassertion event for "Upper Critical Going High" enabled
        Deassertion event for "Upper Non-Critical Going High"
enabled
#
```

Abrufen der Ereignisaktivierungsinformationen für denselben Sensor, jedoch mit Angabe der LU-Nummer und Sensornummer.

```
# clia getsensoreventenable -v fe 0:3

IPM Sentry Shelf Manager Command Line Interpreter

fe: LUN: 0, Sensor # 3 ("Local Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Assertion event mask:   0x0a80
        Assertion event for "Upper Non-Recoverable Going High"
enabled
        Assertion event for "Upper Critical Going High" enabled
       Assertion event for "Upper Non-Critical Going High" enabled
    Deassertion event mask: 0x0a80
        Deassertion event for "Upper Non-Recoverable Going High"
enabled
        Deassertion event for "Upper Critical Going High" enabled
        Deassertion event for "Upper Non-Critical Going High"
enabled
#
```

# getthreshold | threshold

### Syntax:

getthreshold [*IPMB-adresse* [*sensorname* | [*lun*:]*sensornummer*]]
getthreshold board *n* [*sensorname* | [*lun*:]*sensornummer*]]
getthreshold shm *n* [*sensorname* | [*lun*:]*sensornummer*]]

Sie können den Befehl `threshold` auch anstelle von `getthreshold` verwenden.

### Zweck:

Dieser Befehl zeigt die aktuellen Schwellenwerte für die unterstützten Schwellen der angegebenen Sensoren an. Beim Sensor muss es sich um einen Sensor mit Schwellenwert handeln. Es werden sowohl Rohwerte als auch verarbeitete Werte angezeigt. Außerdem werden für jeden Sensor die folgenden Attribute angezeigt:

- IPMB-Adresse des als Eigentümer fungierenden IPM-Controllers
- Sensornummer, Sensorname (Geräte-ID-Zeichenfolge aus dem SDR) und die LU-Nummer, über die auf den Sensor zugegriffen werden kann
- Sensortyp und Ereignis-/Lesetypcode

Mithilfe dieses Befehls können Benutzer die Sensornummer mit der Nummer der logischen Einheit (Logical Unit Number, LUN) qualifizieren, wenn der Zielcontroller Sensoren auf mehreren LU-Nummern unterstützt. Wenn die LU-Nummer fehlt, werden Informationen zu Sensoren mit der angegebenen Sensornummer für alle LU-Nummern angezeigt. *lun* kann den Wert 0, 1 oder 3 haben. (LUN 2 ist reserviert.)

Sensornamen sind nicht durch LU-Nummern qualifiziert, da vorausgesetzt wird, dass Sensornamen innerhalb eines Controllers in der Regel eindeutig sind. Wenn in einem Controller jedoch mehrere Sensoren mit demselben Namen vorhanden sind, werden zu allen Sensoren Informationen angezeigt.

### Beispiele:

Abrufen von Schwellenwerten für einen Temperatursensor `Local Temp` beim IPM-Controller FE.

```
# clia getthreshold -v fe "Local Temp"

IPM Sentry Shelf Manager Command Line Interpreter

fe: LUN: 0, Sensor # 3 ("Local Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
```

```
        Lower Critical Threshold, Raw Data: 0x80, Processed Data:
-128.000000 degrees C
        Upper Non-Critical Threshold, Raw Data: 0x50, Processed
Data: 80.000000 degrees C
        Upper Critical Threshold, Raw Data: 0x50, Processed Data:
80.000000 degrees C
       Upper Non-Recoverable Threshold, Raw Data: 0x50, Processed
Data: 80.000000 degrees C
#
```

Abrufen der Schwellenwertinformationen für denselben Sensor, jedoch mit Angabe der LU-Nummer und Sensornummer.

```
# clia getthreshold -v fe 0:3

IPM Sentry Shelf Manager Command Line Interpreter

fe: LUN: 0, Sensor # 3 ("Local Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
        Lower Critical Threshold, Raw Data: 0x80, Processed Data:
-128.000000 degrees C
        Upper Non-Critical Threshold, Raw Data: 0x50, Processed
Data: 80.000000 degrees C
        Upper Critical Threshold, Raw Data: 0x50, Processed Data:
80.000000 degrees C
       Upper Non-Recoverable Threshold, Raw Data: 0x50, Processed
Data: 80.000000 degrees C
#
```

# help

### Syntax:

help [*befehl* [*unterbefehl*]]

### Zweck:

Dieser Befehl zeigt hilfreiche Informationen zu unterstützten Befehlen und ihrer Syntax an.

### Beispiele:

Abrufen einer Liste von Befehlen und ihrer Syntax.

```
# clia help
IPM Sentry Shelf Manager Command Line Interpreter
Command Line Interface command set:
Parameters are case insensitive
In general:
        IPMB address is hexadecimal ALWAYS.
        All other numbers may be either decimal and hexadecimal (0x notation
        required for hexadecimal numbers)
        -v turns on verbose output

       activate <addr> <fru_id>
       alarm <alarm status/action>
       board [slot_number]
       boardreset <slot number>
       busres force <res>
       busres info [<res>]
       busres lock <res>
       busres query [-v] <res> [<target> [noupdate]]
       busres release <res>
       busres sendbusfree <res> <target>
       busres setowner <res> <target>
       busres unlock <res>
       console [slot_number]
       deactivate <addr> <fru_id>
       debuglevel [<mask>]
       exit
       fans <addr> <fru id>
       fru [<addr> [id=<fru_id> | type=<site_type>]] | [type=<site_type>
           [/<site_number>]]
       frucontrol <addr> <fru_id> <command>
```

```
frudata [<addr>] [<fru id>] [<block number>]
frudata shm <N> [<block number>]
frudata <addr> <fru id> <byte offset> <byte_1> [byte2 .. [byte_16]]
frudatar <addr> <fru id> <file name>
frudataw <addr> <fru id> <file name>
fruinfo <addr> <fru_id>
getfanlevel <addr> <fru_id>
getlanconfig <channel number> <parameter number> | <parameter name>
getpefconfig <parameter name> | <parameter number> [<set selector>]
getsensoreventenable [ <addr> [ [ lun: ]<sensor_id> | <sensor name>  ] ]
getthreshold [ <addr> [ [ lun: ]<sensor id> | <sensor name> ] ]
help [<command>]
ipmc [<addr>]
localaddress
minfanlevel [<min fan level>]
poll
quit
sel [clear] [ <addr> [ <number of items> [<number of first item>] ] ]
sel info [<addr>]
sensor [ <addr> [ [ lun: ]<sensor id> | <sensor name> ] ]
sensordata [ <addr> [ [ lun: ]<sensor id> | <sensor name> ] ]
sensorread <addr> [ lun: ]<sensor id>
session
setextracted <addr> <fru_id>
setfanlevel <addr> <fru_id> <state>
setlanconfig <channel number> <parameter number> | parameter name
             <parameters ...>
setlocked <addr> <fru_id> <value>
setpefconfig <parameter name> | <parameter number> [<set selector>]
             <parameters ...>
setsensoreventenable <addr> [ lun: ]<sensor_id> | <sensor name> global
             [assertion_events [deassertion_events]]
setthreshold <addr> [ lun: ]<sensor_id> | <sensor name> unc | uc | unr
             | lnc | lc | lnr [-r] value
shelf <parameters>
shelfaddress ["<shelf address>"]
shmstatus
showunhealthy
switchover
threshold [ <addr> [ [ lun: ]<sensor id> | <sensor name> ] ]
user [<user id>]
user add <user id> <user name> <flags> <privilege level> <password>
 user channel <user id> <channel number> <flags> <privilege level>
 user delete <user id>
 user delete <user id>
 user enable <user id> 1|0
 user name <user id> <user name
 user passwd <user id> <user password>
 version
```

```
# clia help shelf pwrreorder
IPM Sentry Shelf Manager Command Line Interpreter
        Set the Power Order
        PwrReorder <addr1> <fru_id1> before/after <addr2> <fru_id2>
#
```

Abrufen von Hilfe zu einem bestimmten Befehl.


```
# clia help shelf pwrreorder
IPM Sentry Shelf Manager Command Line Interpreter
        Set the Power Order
        PwrReorder <addr1> <fru_id1> before/after <addr2> <fru_id2>
#
```


# ipmc

### Syntax:

ipmc [-v] [*IPMB-adresse*]
ipmc board *n*
ipmc fan_tray *n*

### Zweck:

Dieser Befehl zeigt Informationen zum IPM-Controller unter der angegebenen Adresse bzw. zu allen IPM-Controllern an, die dem Shelf Manager bekannt sind, wenn für *IPMB-adresse* kein Wert angegeben ist.

Im Standardmodus werden für den IPM-Controller die folgenden Informationen angezeigt:

- IPMB-Adresse des Controllers in Form von zwei hexadezimalen Ziffern
- Entitäts-ID und Entitätsinstanz für den IPM-Controller
- Maximal zulässige FRU-Geräte-ID für den IPM-Controller
- PICMG-Erweiterungsversion; diese Version muss für PICMG 3.0-kompatible IPM-Controller 2.0 lauten

Aktueller Hot-Swap-Status, vorheriger Hot-Swap-Status und Ursache der letzten Statusänderung für FRU-Gerät 0 des IPM-Controllers (das für den IPM-Controller selbst steht). Die Hot-Swap-Statustypen M0 bis M7 sind in der Spezifikation PICMG 3.0 folgendermaßen definiert:

- M0 – Nicht installiert

- M1 – Inaktiv
- M2 – Aktivierungsanforderung
- M3 – Aktivierung wird durchgeführt
- M4 – FRU aktiv
- M5 – Deaktivierungsanforderung
- M6 – Deaktivierung wird durchgeführt
- M7 – Kommunikation unterbrochen

Im ausführlichen Modus werden für den IPM-Controller die folgenden zusätzlichen Informationen angezeigt:

- Informationen, die vom IPMI-Befehl `Get Device ID` zurückgegeben werden, z. B. Hersteller-ID, Produkt-ID, Geräte-ID, Version der Geräte-Firmware und unterstützte IPMI-Version
- Geräte-ID-Zeichenfolge aus dem Controller-SDR
- Attribut der Betriebszustand-Benachrichtigung aus dem Controller-SDR als hexadezimale Zahl
- Globales Initialisierungsattribut aus dem Controller-SDR als hexadezimale Zahl
- Attribut der Gerätefunktionen aus dem Controller-SDR als hexadezimale Zahl
- Hinweis, ob der Controller Geräte-SDRs bereitstellt
- Maske der unterstützten Funktionen mit einer Texterklärung der einzelnen Bits
- Liste der E-Keying-Anschlüsse mit dem jeweiligen Status (Aktiviert/Deaktiviert)

**Beispiele:**

Abrufen von Informationen zum IPM-Controller unter der Adresse 9C.

```
# clia ipmc 9c
IPM Sentry Shelf Manager Command Line Interpreter
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
#
```

Abrufen von ausführlichen Informationen zum IPM-Controller unter der Adresse 9C.

```
# clia ipmc -v 9c
IPM Sentry Shelf Manager Command Line Interpreter
9c: Entity: (0xd0, 0x0) Maximum FRU device ID: 0x08
    PICMG Version 2.0
    Hot Swap State: M4 (Active), Previous: M3 (Activation In
Process), Last State Change Cause: Normal State Change (0x0)
    Device ID: 0x00, Revision: 0, Firmware: 1.01, IPMI ver 1.5
```

```
    Manufacturer ID: 00315a (PICMG), Product ID: 0000, Auxiliary
Rev: 01ac10ac
    Device ID String: "IPM Sentry 6"
    Global Initialization: 0x0, Power State Notification: 0x0,
Device Capabilities: 0x29
    Controller provides Device SDRs
    Supported features: 0x29
        "Sensor Device" "FRU Inventory Device" "IPMB Event
Generator"
#
```

# localaddress

**Syntax:**

`localaddress`

**Zweck:**

Dieser Befehl zeigt die IPMB-Adresse des aktuellen Shelf Managers basierend auf seiner Hardwareadresse an (nicht anhand seiner generischen BMC-Adresse 0x20). Diese Adressen sind bei redundanten Shelf Managern unterschiedlich (während die BMC-Adresse gemeinsam genutzt wird, also identisch ist).

**Beispiel:**

```
# clia localaddress
IPM Sentry Shelf Manager Command Line Interpreter
Local IPMB Address = 0xFC
#
```

# minfanlevel

**Syntax:**

`minfanlevel [`*einstellung*`]`

### Zweck:

Dieser Befehl zeigt die minimale Lüftereinstellung an bzw. nimmt sie vor. Unter normalen Bedingungen verringert der Lüftungsverwaltungsalgorithmus die Einstellung der Lüfter des Systems allmählich, solange keine außergewöhnlichen Wärmebedingungen herrschen. Der Lüftungsverwaltungsalgorithmus verringert die Lüftereinstellung jedoch nicht auf einen Wert unterhalb der minimalen Einstellung, die durch den Konfigurationsparameter `MIN_FAN_LEVEL` oder diesen Befehl vorgegeben wird.

Der Standardwert für die minimale Lüftereinstellung ist 1. Wenn Sie diese Einstellung auf einen höheren Wert setzen, wird dadurch nicht verhindert, dass die Lüftereinstellung mithilfe des Befehls `clia setfanlevel` oder des ATCA-Befehls `SetFanLevel`, der per RMCP angewendet werden kann, auf einen niedrigeren Wert gesetzt wird. Die minimale Lüftereinstellung gilt nur für die automatische Verwaltung der Lüftereinstellung durch das Dienstprogramm für die Lüftungsverwaltung.

Dieser Befehl zeigt ohne Parameter die aktuelle minimale Lüftereinstellung an.

Wenn Sie diesen Befehl mit einer Ganzzahl als Parameter verwenden, wird die minimale Lüftereinstellung auf den im Parameter angegebenen Wert gesetzt.

### Beispiel:

```
# clia minfanlevel 3
IPM Sentry Shelf Manager Command Line Interpreter
Minimal Fan Level is set to 3

# clia minfanlevel
IPM Sentry Shelf Manager Command Line Interpreter
Minimal Fan Level is 3
#
```

# sel

### Syntax:

`sel [-v]` [*IPMB-adresse* [*datensatzanzahl* [*starteintrag*]]]
`sel clear` [*IPMB-adresse*]
`sel info` [*IPMB-adresse*]

Sie können *IPMB-adresse* durch die Abkürzungen `board` *n* oder `shm` *n* ersetzen.

### Zweck:

Dieser Befehl zeigt den Inhalt des Systemereignisprotokolls (SEL) für den angegebenen IPM-Controller an (standardmäßig unter der IPMB-Adresse 20h). Sie können den optionalen Parameter *datensatzanzahl* verwenden, um anzugeben, wie viele Datensätze ab der Datensatznummer *starteintrag* im SEL angezeigt werden. Beim optionalen Parameter *starteintrag* handelt es sich um die Eintragsnummer des ersten SEL-Datensatzes, der relativ zum Anfang des SEL angezeigt werden soll. Die Werte für *datensatzanzahl* und *starteintrag* müssen im Bereich von 1 bis zur Gesamtzahl der Datensätze im SEL liegen. Der Standardwert des optionalen Parameters *starteintrag* ist 1. Der Parameter *starteintrag* ist unabhängig vom Feld `RecordID` (Datensatz-ID) des SEL-Datensatzes.

Für jeden SEL-Datensatz werden die folgenden Informationsfelder angezeigt:

- Datensatz-ID
- Datensatztyp (momentan werden nur Ereignisse unterstützt, für die das Wort `Event` (Ereignis) angezeigt wird.
- Zeitstempel (für Datensätze mit Zeitstempel)
- Quelladressenparameter: IPMB-Adresse, LU-Nummer und Kanalnummer
- Typ und Nummer des Sensors, der das Ereignis erzeugt hat
- Ereignis-/Lesetypcode
- Drei Byte an Ereignisdaten im Rohformat und (falls verfügbar) im verarbeiteten Format

Der Befehl `sel clear` löscht das SEL für den angegebenen IPM-Controller (standardmäßig unter der IPMB-Adresse 20h).

Die Option `-v` erhöht die Benutzerfreundlichkeit der Ausgabe von SEL-Einträgen.

### Beispiele:

Lesen des SEL auf dem Shelf Manager.

```
# clia sel info
IPM Sentry Shelf Manager Command Line Interpreter
20: SEL version: 1.5
    Number of log entries: 43
    Free space: 15680 bytes
    Last addition timestamp: Nov 19 17:12:47 2003
    Last erase timestamp: Oct 31 23:59:59 2003
    Supported operations: 0x0f

# clia sel 20 5
IPM Sentry Shelf Manager Command Line Interpreter
```

```
0x0027: Event: at Nov 19 17:12:42 2003; from:(0x9c,0,0);
sensor:(0xf0,0); event:0x6f(asserted): HotSwap: FRU 0 M4->M6,
Cause=0x1
0x0028: Event: at Nov 19 17:12:42 2003; from:(0x9c,0,0);
sensor:(0xf0,0); event:0x6f(asserted): HotSwap: FRU 0 M6->M1,
Cause=0x0
0x0029: Event: at Nov 19 17:12:46 2003; from:(0x9c,0,0);
sensor:(0xf0,0); event:0x6f(asserted): HotSwap: FRU 0 M1->M2,
Cause=0x2
0x002A: Event: at Nov 19 17:12:46 2003; from:(0x9c,0,0);
sensor:(0xf0,0); event:0x6f(asserted): HotSwap: FRU 0 M2->M3,
Cause=0x1
0x002B: Event: at Nov 19 17:12:47 2003; from:(0x9c,0,0);
sensor:(0xf0,0); event:0x6f(asserted): HotSwap: FRU 0 M3->M4,
Cause=0x0

# clia sel b4 5
IPM Sentry Shelf Manager Command Line Interpreter
0x00A4: Event: at Nov 19 01:24:25 2003; from:(0x20,0,0);
sensor:(0x02,4); event:0x1(asserted): "Lower Non-Critical",
Threshold: 0xb3, Reading: 0xb3
0x00B8: Event: at Nov 19 00:04:11 2003; from:(0x20,0,0);
sensor:(0x02,4); event:0x1(asserted): "Lower Non-Critical",
Threshold: 0xb3, Reading: 0xb3
0x00CC: Event: at Nov 19 00:36:32 2003; from:(0x20,0,0);
sensor:(0x02,7); event:0x1(asserted): "Lower Non-Critical",
Threshold: 0xae, Reading: 0x94
0x00E0: Event: at Nov 19 00:36:32 2003; from:(0x20,0,0);
sensor:(0x02,7); event:0x1(asserted): "Lower Critical",
Threshold: 0xac, Reading: 0x94
0x00F4: Event: at Nov 19 00:02:37 2003; from:(0x20,0,0);
sensor:(0x01,2); event:0x1(asserted): "Upper Critical",
Threshold: 0x13, Reading: 0x1c

# clia sel -v board 3 5
IPM Sentry Shelf Manager Command Line Interpreter
0x00A4: Event: at: Nov 19 01:24:25 2003; from IPM Controller: 0x20,
LUN: 0, Channel: 0
    "Voltage" (0x02) sensor # 4
    "Threshold" (0x01) event Asserted
    "Lower Non-Critical Going Low"
    Reading value: 0xb3
    Threshold value: 0xb3
0x00B8: Event: at: Nov 19 00:04:11 2003; from IPM Controller: 0x20,
LUN: 0, Channel: 0
    "Voltage" (0x02) sensor # 4
```

```
    "Threshold" (0x01) event Asserted
    "Lower Non-Critical Going Low"
    Reading value: 0xb3
    Threshold value: 0xb3
0x00CC: Event: at: Nov 19 00:36:32 2003; from IPM Controller: 0x20,
LUN: 0, Channel: 0
    "Voltage" (0x02) sensor # 7
    "Threshold" (0x01) event Asserted
    "Lower Non-Critical Going Low"
    Reading value: 0x94
    Threshold value: 0xae
0x00E0: Event: at: Nov 19 00:36:32 2003; from IPM Controller: 0x20,
LUN: 0, Channel: 0
    "Voltage" (0x02) sensor # 7
    "Threshold" (0x01) event Asserted
    "Lower Critical Going Low"
    Reading value: 0x94
    Threshold value: 0xac
0x00F4: Event: at: Nov 19 00:02:37 2003; from IPM Controller: 0x20,
LUN: 0, Channel: 0
    "Temperature" (0x01) sensor # 2
    "Threshold" (0x01) event Asserted
    "Upper Critical Going High"
    Reading value: 0x1c
    Threshold value: 0x13
#
```

Abrufen von fünf SEL-Einträgen aus Eintrag # 15 (0x0f).

```
# clia sel 20 5 15
IPM Sentry Shelf Manager Command Line Interpreter
0x000F: Event: at Nov 19 16:49:21 2003; from:(0x20,0,0);
sensor:(0xf0,3); event:0x6f(asserted): HotSwap: FRU 2 M2->M3,
Cause=0x1
0x0010: Event: at Nov 19 16:49:22 2003; from:(0x20,0,0);
sensor:(0xf0,2); event:0x6f(asserted): HotSwap: FRU 1 M2->M3,
Cause=0x1
0x0011: Event: at Nov 19 16:49:22 2003; from:(0x20,0,0);
sensor:(0xf0,2); event:0x6f(asserted): HotSwap: FRU 1 M3->M4,
Cause=0x0
```

```
0x0012: Event: at Nov 19 16:49:22 2003; from:(0xfc,0,0);
sensor:(0xf0,0); event:0x6f(asserted): HotSwap: FRU 0 M3->M4,
Cause=0x0
0x0013: Event: at Nov 19 16:49:22 2003; from:(0x20,0,0);
sensor:(0xf0,3); event:0x6f(asserted): HotSwap: FRU 2 M3->M4,
Cause=0x0
#
```

Löschen des SEL.

```
# clia sel clear
IPM Sentry Shelf Manager Command Line Interpreter
SEL clear: issued successfully
    SEL clearing completed
# clia sel
IPM Sentry Shelf Manager Command Line Interpreter
SEL is empty
#
```

# sensor

### Syntax:

sensor [-v] [*IPMB-adresse* [*sensorname* | [*lun*:]*sensornummer*]]
sensor [-v] board *n* [*sensorname* | [*lun*:]*sensornummer*]]
sensor [-v] shm *n* [*sensorname* | [*lun*:]*sensornummer*]]

### Zweck:

Dieser Befehl zeigt Informationen zu bestimmten Sensoren an. Der Zielsensor wird über die IPMB-Adresse seines IPM-Controllers und nach der Sensornummer bzw. dem Sensornamen ausgewählt (Geräte-ID-Zeichenfolge aus dem Sensor-SDR in doppelten Anführungszeichen). Wenn weder der Sensorname noch die Sensornummer angegeben ist, werden Informationen zu allen Sensoren des entsprechenden IPM-Controllers angezeigt. Wenn keine Parameter angegeben sind, werden Informationen zu allen bekannten Sensoren angezeigt.

Mithilfe dieses Befehls können Benutzer die Sensornummer mit der Nummer der logischen Einheit (Logical Unit Number, LUN) qualifizieren, wenn der Zielcontroller Sensoren auf mehreren LU-Nummern unterstützt. Wenn die LU-Nummer fehlt, werden Informationen zu Sensoren mit der angegebenen Sensornummer für alle LU-Nummern angezeigt. *lun* kann den Wert 0, 1 oder 3 haben (LUN 2 ist reserviert).

Sensornamen sind nicht durch LU-Nummern qualifiziert, da vorausgesetzt wird, dass Sensornamen innerhalb eines Controllers in der Regel eindeutig sind. Wenn in einem Controller jedoch mehrere Sensoren mit demselben Namen vorhanden sind, werden zu allen Sensoren Informationen angezeigt.

Im Standardmodus werden für jeden Sensor die folgenden Informationen angezeigt:

- IPMB-Adresse des als Eigentümer fungierenden IPM-Controllers
- Sensornummer, Sensorname (Geräte-ID-Zeichenfolge aus dem SDR) und die LU-Nummer, über die auf den Sensor zugegriffen werden kann
- Sensortyp und Ereignis-/Lesetypcode
- Entitäts-ID, Entitätsinstanz der dazugehörigen Entität (FRU-Geräte-ID, wenn der Sensor einer FRU zugeordnet ist)

Die folgenden Informationen werden für den Sensor nur im ausführlichen Modus angezeigt (Informationen zu diesen Attributen finden Sie in der IPMI-Spezifikation):

- Assertion Mask
- Deassertion Mask
- Einstellbare/lesbare Maske für Sensorstatus (bei einem diskreten Sensor) oder Schwellenwerte (bei einem Sensor mit Schwellenwert)

Die folgenden Informationen werden für Sensoren mit Schwellenwerten nur im ausführlichen Modus angezeigt:

- Sensoreinheiten: Basiswert und geänderter Wert
- Einheitenprozentsatz, Modifizierer und Rate
- Analogformat und Flags
- Linearisierungsparameter, Koeffizienten M, B, K1, K2
- Toleranz- und Genauigkeitskoeffizienten
- Werte für Nennwert, normales Maximum, normales Minimum, Maximum und Minimum
- Obere Schwellenwerte: nicht kritisch, kritisch und nicht wiederherstellbar
- Untere Schwellenwerte: nicht kritisch, kritisch und nicht wiederherstellbar
- Hysteresewerte: positiv und negativ

### Beispiele:

Abrufen von Standardinformationen zum Sensor `FAN 4` für den IPM-Controller FE.

```
# clia sensor fe "FAN 4"
IPM Sentry Shelf Manager Command Line Interpreter
fe: LUN: 0, Sensor # 14 ("FAN 4")
    Type: Threshold (0x01), "Fan" (0x04)
    Belongs to entity: (0xd0, 0) [FRU # 0]
#
```

Abrufen von ausführlichen Informationen zum Sensor 2 für den IPM-Controller 9C.

```
# clia sensor -v 9c 2
IPM Sentry Shelf Manager Command Line Interpreter
9c: LUN: 0, Sensor # 2 ("emulated temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Belongs to entity: (0xd0, 0) [FRU # 0]
    Assertion Mask: 0x7a95
        Lower Non-Critical Going Low
        Lower Critical Going Low
        Lower Non-Recoverable Going Low
        Upper Non-Critical Going High
        Upper Critical Going High
        Upper Non-Recoverable Going High
        Upper non-critical threshold is comparison returned
        Upper critical threshold is comparison returned
        Upper non-recoverable threshold comparison is returned
    Deassertion Mask: 0x7a95
        Lower Non-Critical Going Low
        Lower Critical Going Low
        Lower Non-Recoverable Going Low
        Upper Non-Critical Going High
        Upper Critical Going High
        Upper Non-Recoverable Going High
        Upper non-critical threshold is comparison returned
        Upper critical threshold is comparison returned
        Upper non-recoverable threshold comparison is returned
    Settable / Readable Mask: 0x3f3f
        Lower Non-Critical Threshold is Readable
        Lower Critical Threshold is Readable
        Lower Non-Recoverable Threshold is Readable
        Upper Non-Critical Threshold is Readable
        Upper Critical Threshold is Readable
        Upper Non-Recoverable Threshold is Readable
        Lower Non-Critical Threshold is Settable
        Lower Critical Threshold is Settable
        Lower Non-Recoverable Threshold is Settable
        Upper Non-Critical Threshold is Settable
        Upper Critical Threshold is Settable
        Upper Non-Recoverable Threshold is Settable
    Unit Percentage: OFF (0), Unit Modifier: none (0), Unit Rate:
none (0)
    Analog Format: 2's complement (signed) (2)
    Base Unit: degrees C (1), Modifier Unit: unspecified (0)
    Linearization: linear (0), M = 1, B = 0, K1 = 0, K2 = 0
```

```
    Tolerance = 0, Accuracy = 0, Accuracy EXP = 0
    Analog Flags: 0x0
    Nominal: 0 (0x00), Normal max: 0 (0x00), Normal min: 0 (0x00)
    Sensor max: 127 (0x7f), Sensor min: 128 (0x80)
    Upper Thresholds:
      Non-Critical: 70 (0x46) Critical: 80 (0x50) Non-Recoverable:
90 (0x5a)
    Lower Thresholds:
       Non-Critical: 3 (0x03) Critical: 0 (0x00) Non-Recoverable:
251 (0xfb)
    Hysteresis:
        Positive: 2 (0x02), Negative 2 (0x02)
#
```

Wie oben, jedoch mit expliziter Angabe der LU-Nummer für den Sensor.

```
# clia sensor -v 9c 0:2
IPM Sentry Shelf Manager Command Line Interpreter
9c: LUN: 0, Sensor # 2 ("emulated temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Belongs to entity: (0xd0, 0) [FRU # 0]
    Assertion Mask: 0x7a95
        Lower Non-Critical Going Low
        Lower Critical Going Low
        Lower Non-Recoverable Going Low
        Upper Non-Critical Going High
        Upper Critical Going High
        Upper Non-Recoverable Going High
        Upper non-critical threshold is comparison returned
        Upper critical threshold is comparison returned
        Upper non-recoverable threshold comparison is returned
    Deassertion Mask: 0x7a95
        Lower Non-Critical Going Low
        Lower Critical Going Low
        Lower Non-Recoverable Going Low
        Upper Non-Critical Going High
        Upper Critical Going High
        Upper Non-Recoverable Going High
        Upper non-critical threshold is comparison returned
        Upper critical threshold is comparison returned
        Upper non-recoverable threshold comparison is returned
    Settable / Readable Mask: 0x3f3f
        Lower Non-Critical Threshold is Readable
        Lower Critical Threshold is Readable
        Lower Non-Recoverable Threshold is Readable
```

```
        Upper Non-Critical Threshold is Readable
        Upper Critical Threshold is Readable
        Upper Non-Recoverable Threshold is Readable
        Lower Non-Critical Threshold is Settable
        Lower Critical Threshold is Settable
        Lower Non-Recoverable Threshold is Settable
        Upper Non-Critical Threshold is Settable
        Upper Critical Threshold is Settable
        Upper Non-Recoverable Threshold is Settable
    Unit Percentage: OFF (0), Unit Modifier: none (0), Unit Rate:
none (0)
    Analog Format: 2's complement (signed) (2)
    Base Unit: degrees C (1), Modifier Unit: unspecified (0)
    Linearization: linear (0), M = 1, B = 0, K1 = 0, K2 = 0
    Tolerance = 0, Accuracy = 0, Accuracy EXP = 0
    Analog Flags: 0x0
    Nominal: 0 (0x00), Normal max: 0 (0x00), Normal min: 0 (0x00)
    Sensor max: 127 (0x7f), Sensor min: 128 (0x80)
    Upper Thresholds:
      Non-Critical: 70 (0x46) Critical: 80 (0x50) Non-Recoverable:
90 (0x5a)
    Lower Thresholds:
       Non-Critical: 3 (0x03) Critical: 0 (0x00) Non-Recoverable:
251 (0xfb)
    Hysteresis:
        Positive: 2 (0x02), Negative 2 (0x02)
#
```

# sensordata

### Syntax:

sensordata [*IPMB-adresse* [*sensorname* | [*lun*:]*sensornummer*]]
sensordata [-v] board *n* [*sensorname* | [*lun*:]*sensornummer*]]
sensordata [-v] shm *n* [*sensorname* | [*lun*:]*sensornummer*]]

### Zweck:

Dieser Befehl zeigt den tatsächlichen Wert des angegebenen Sensors (für einen Sensor mit Schwellenwert) oder die momentan bestätigten Statuszustände an (für einen diskreten Sensor). Der Zielsensor wird über die IPMB-Adresse seines IPM-Controllers und nach der Sensornummer bzw. dem Sensornamen ausgewählt

(Geräte-ID-Zeichenfolge aus dem Sensor-SDR in doppelten Anführungszeichen). Wenn weder der Sensorname noch die Sensornummer angegeben ist, werden Werte aller Sensoren des entsprechenden IPM-Controllers angezeigt. Wenn keine Parameter angegeben sind, werden Werte aller bekannten Sensoren angezeigt.

Mithilfe dieses Befehls können Benutzer die Sensornummer mit der Nummer der logischen Einheit (Logical Unit Number, LUN) qualifizieren, wenn der Zielcontroller Sensoren auf mehreren LU-Nummern unterstützt. Wenn die LU-Nummer fehlt, werden Informationen zu Sensoren mit der angegebenen Sensornummer für alle LU-Nummern angezeigt. *lun* kann den Wert 0, 1 oder 3 haben. (LUN 2 ist reserviert.)

Sensornamen sind nicht durch LU-Nummern qualifiziert, da vorausgesetzt wird, dass Sensornamen innerhalb eines Controllers in der Regel eindeutig sind. Wenn in einem Controller jedoch mehrere Sensoren mit demselben Namen vorhanden sind, werden zu allen Sensoren Informationen angezeigt.

Für jeden Sensor werden die folgenden Informationen angezeigt:

- IPMB-Adresse des als Eigentümer fungierenden IPM-Controllers
- Sensornummer, Sensorname (Geräte-ID-Zeichenfolge aus dem SDR) und die LU-Nummer, über die auf den Sensor zugegriffen werden kann
- Sensortyp und Ereignis-/Lesetypcode
- Sensorwert (für Sensoren mit Schwellenwert) oder die Maske momentan bestätigter Statuszustände (für diskrete Sensoren) in Rohform
- Status bei Erreichen des Schwellenwerts in hexadezimaler Form und mit Decodierung

Der Wert bzw. die bestätigten Statuszustände werden sowohl in Rohform als auch in verarbeiteter Form angezeigt. In verarbeiteter Form werden die analogen Werte gemäß M, B und R konvertiert und zusammen mit dem Einheitennamen (z. B. 27 Grad) angezeigt. Der diskrete Wert wird je nach Ereignis-/Lesecodetyp mit einer entsprechenden Anmerkung versehen (für den Ereignis-/Lesecode 2 wird Status 0 z. B. als `Transition to Idle` angezeigt).

### Beispiele:

Abrufen von Sensordatenwerten für einen Temperatursensor `Local Temp` beim IPM-Controller FE.

```
# clia sensordata FE "Local Temp"
IPM Sentry Shelf Manager Command Line Interpreter
fe: LUN: 0, Sensor # 3 ("Local Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Status: 0xc0
        All event messages enabled from this sensor
        Sensor scanning enabled
        Initial update completed
```

```
    Raw data: 22 (0x16)
    Processed data: 22.000000 degrees C
    Status: 0x00
```

Abrufen von Sensordatenwerten für einen diskreten (Hot-Swap-) Sensor (#0) beim IPM-Controller 9C.

```
# clia sensordata 9c 0
IPM Sentry Shelf Manager Command Line Interpreter
9c: LUN: 0, Sensor # 0 ("FRU 0 HOT_SWAP")
    Type: Discrete (0x6f), "Hot Swap" (0xf0)
    Status: 0xc0
        All event messages enabled from this sensor
        Sensor scanning enabled
        Initial update completed
    Sensor reading: 0x00
    Current State Mask 0x0010
```

Abrufen von Sensordatenwerten für denselben Sensor, jedoch mit expliziter Angabe der LU-Nummer.

```
# clia sensordata 9c 0:0
IPM Sentry Shelf Manager Command Line Interpreter
9c: LUN: 0, Sensor # 0 ("FRU 0 HOT_SWAP")
    Type: Discrete (0x6f), "Hot Swap" (0xf0)
    Status: 0xc0
        All event messages enabled from this sensor
        Sensor scanning enabled
        Initial update completed
    Sensor reading: 0x00
    Current State Mask 0x0010
```

# sensorread

### Syntax:

`sensorread` *IPMB-adresse* `[`*lun*`:]`*sensornummer*

## Zweck:

Dieser Befehl zeigt den Rohwert des angegebenen Sensors an. Der einzige Unterschied zwischen den Befehlen `sensorread` und `sensordata` besteht darin, dass der Befehl `sensorread` das Vorhandensein des IPM-Zielcontrollers bzw. die Gültigkeit der Sensornummer nicht prüft, sondern nur die Anforderung `Get Sensor Reading` direkt per IPMB sendet. Dieser Befehl ruft den SDR des Sensors nicht ab und kann die erfassten Daten daher nicht verarbeiten.

Mithilfe dieses Befehls können Benutzer die Sensornummer mit der Nummer der logischen Einheit (Logical Unit Number, LUN) qualifizieren, wenn der Zielcontroller Sensoren auf mehreren LU-Nummern unterstützt. Wenn die LU-Nummer fehlt, wird die LU-Nummer 0 verwendet. *lun* kann den Wert 0, 1 oder 3 haben (LUN 2 ist reserviert).

Für jeden Sensor werden die folgenden Informationen angezeigt:

- IPMB-Adresse des als Eigentümer fungierenden IPM-Controllers
- Sensornummer, Sensorname (Geräte-ID-Zeichenfolge aus dem SDR) und die LU-Nummer, über die auf den Sensor zugegriffen werden kann
- Sensortyp und Ereignis-/Lesetypcode
- Sensorwert (für Sensoren mit Schwellenwert) oder die Maske momentan bestätigter Statuszustände (für diskrete Sensoren) in Rohform

## Beispiele:

Abrufen von Sensordatenwerten für Sensor 4 auf dem IPM-Controller FC. Beachten Sie, dass der Befehl `sensorread` nur nicht verarbeitete Sensorwerte bereitstellt. Sehen Sie sich auch das Befehlsbeispiel mit der explizit angegebenen LU-Nummer an.

```
# clia sensordata fc 4
IPM Sentry Shelf Manager Command Line Interpreter
fc: LUN: 0, Sensor # 4 ("3.3STBY voltage")
    Type: Threshold (0x01), "Voltage" (0x02)
    Status: 0xc0
        All event messages enabled from this sensor
        Sensor scanning enabled
        Initial update completed
    Raw data: 193 (0xc1)
    Processed data: 3.396800 Volts
    Status: 0x00

# clia sensorread fc 4
IPM Sentry Shelf Manager Command Line Interpreter
fc: LUN: 0, Sensor # 4
    Raw data: 193 (0xc1)
```

```
    Status: 0xc0
        All event messages enabled from this sensor
        Sensor scanning enabled
        Initial update completed
    Threshold Sensor Status: 0x00
    Discrete Sensor Current State Mask 0x0000

# clia sensorread fc 0:4
IPM Sentry Shelf Manager Command Line Interpreter
fc: LUN: 0, Sensor # 4
    Raw data: 193 (0xc1)
    Status: 0xc0
        All event messages enabled from this sensor
        Sensor scanning enabled
        Initial update completed
    Threshold Sensor Status: 0x00
    Discrete Sensor Current State Mask 0x0000
#
```

# session

### Syntax:

`session`

### Zweck:

Dieser Befehl zeigt Informationen zu aktiven RMCP-Sitzungen an. Die Informationen enthalten die folgenden Elemente:

- Maximal zulässige Anzahl an Sitzungen und Anzahl an momentan aktiven Sitzungen
- Für jede momentan aktive Sitzung:
  - Zugriffsroutine für Sitzung
  - Benutzer-ID und Benutzername, die/der für die Aktivierung der Sitzung verwendet wird
  - Maximal zulässige Berechtigungsebene für Sitzung
  - IPMI-Kanalnummer und -typ
  - Bei LAN-Sitzungen die IP-Adresse und Portnummer der gleichgeordneten Einheiten (Peers)

**Beispiel:**

```
# clia session
IPM Sentry Shelf Manager Command Line Interpreter
32 sessions possible, 2 sessions currently active
Session: 1
   User: ID 1, Name: ""; Privilege Level: "Administrator"
   Channel: 1 ("LAN_802_3"); Peer IP address: 172.16.2.203, Port:
1764
Session: 2
   User: ID 1, Name: ""; Privilege Level: "Administrator"
   Channel: 1 ("LAN_802_3"); Peer IP address: 172.16.2.203, Port:
1765
#
```

# setextracted

**Syntax:**

setextracted *IPMB-adresse fru-id*

**Zweck:**

Dieser Befehl benachrichtigt den Shelf Manager darüber, dass die angegebene FRU aus dem Shelf entfernt wurde. Wenn die angegebene FRU den Status M7 aufweist, ändert der Shelf Manager den Status in M0 (FRU nicht installiert).

**Beispiel:**

```
# clia setextracted 9c 0
IPM Sentry Shelf Manager Command Line Interpreter
Set FRU extracted state successfully
#
```

# setfanlevel

### Syntax:

setfanlevel *IPMB-adresse fru-id einstellung*
setfanlevel fan_tray *n einstellung*
setfanlevel all *einstellung*

### Zweck:

Dieser Befehl legt die neue Einstellung des Lüfters fest, der von der in den Befehlsparametern angegebenen FRU gesteuert wird. Die niedrigste Einstellung ist 1, und die höchste Einstellung ist 15.

Wenn Sie diesen Befehl mit dem Kennzeichner `all` verwenden, wird versucht, auf alle bekannten Lüfter des Shelfs dieselbe Einstellung anzuwenden.

### Beispiele:

Setzen der Lüftereinstellung für den Lüfter von FRU #2 unter der IPMB-Adresse 0x20 auf den Wert 5.

```
# clia setfanlevel 20 2 5

IPM Sentry Shelf Manager Command Line Interpreter

20: FRU # 2 Set Fan Level to: 5
#
```

Setzen der Lüftereinstellung für alle bekannten Lüfter des Shelfs auf den Wert 4:

```
# clia setfanlevel all 4

IPM Sentry Shelf Manager Command Line Interpreter

72: FRU # 0 Set Fan Level to: 4
76: FRU # 0 Set Fan Level to: 4
#
```

# setfruledstate

## Syntax:

`setfruledstate` *IPMB-adresse fru-id LedId*|`ALL` *LedOp* [*LedFarbe*]

*LedOp* = `ON` | `OFF` | `LOCAL` | `BLINK` <*ZeitEin*> <*ZeitAus*> | `TEST` <*ZeitEin*>
*LedColor* = `BLUE` | `RED` | `GREEN` | `AMBER` | `ORANGE` | `WHITE` | `NONE` | *zahl*

## Zweck:

Mithilfe dieses Befehls können Benutzer den Status einer bestimmten LED bzw. aller LEDs einer FRU festlegen.

Das erste Argument *IPMB-adresse* ist die IPMB-Adresse eines IPM-Controllers. Das zweite Argument *fru-id* ist die FRU-Geräte-ID. Das dritte Argument kann entweder eine LED-ID (numerischer Wert) oder `ALL` sein. Im letzten Fall gilt die angegebene Operation für alle LEDs.

Das Argument *LedOp* steht für die Operation, die basierend auf der Spezifikation PICMG 3.0 auf die FRUs angewendet wird. Die Operationen sind wie folgt definiert:

- ON – LED aktivieren
- OFF – LED deaktivieren
- LOCAL – Zur lokalen Steuerung der LED zurückkehren
- BLINK – LED blinken lassen, wobei sie für den Zeitraum *ZeitEin* (Millisekunden) eingeschaltet und dann für den Zeitraum *ZeitAus* (Millisekunden) ausgeschaltet wird
- TEST – Lampentest für einen Zeitraum von *ZeitEin* (Millisekunden) durchführen

Bei der Operation TEST muss der Zeitraum für *ZeitEin* weniger als 12 800 ms (12,8 s) betragen. Bei der Operation BLINK müssen die Werte für *ZeitEin* und *ZeitAus* im Bereich 10 bis 2500 ms liegen.

Der optionale Parameter *LedFarbe* steht für eine Farbe, die entweder mithilfe eines symbolischen Namens oder eines Dezimalwerts angegeben werden kann. Symbolische Namen für Farben entsprechen gemäß der Spezifikation PICMG 3.0 Dezimalwerten, die unten aufgeführt sind. (Wenn der Parameter nicht angegeben ist, wird die Standardfarbe der LED verwendet.)

- BLUE = 1
- RED = 2
- GREEN = 3
- AMBER = 4
- ORANGE = 6

- NONE = 14 (Farbe nicht ändern)

### Beispiele:

Deaktivieren von LED #1 für FRU #0 des IPM-Controllers unter der IPMB-Adresse 20h.

```
# clia setfruledstate  20 0 1 OFF
IPM Sentry Shelf Manager Command Line Interpreter

    Setting FRU's led state completed successfully, status = 0x0
#
```

Aktivieren der lokalen Steuerung von LED #1 für FRU #0 des IPM-Controllers unter der IPMB-Adresse 20h.

```
# clia setfruledstate  20 0 1 LOCAL
IPM Sentry Shelf Manager Command Line Interpreter

    Setting FRU's led state completed successfully, status = 0x0
#
```

Aktivieren des Blinkens von LED #1 für FRU #0 des IPM-Controllers unter der IPMB-Adresse 20h. Das Blinken erfolgt in der Standardfarbe. Die Einschaltdauer beträgt 100 ms, und die Ausschaltdauer beträgt 200 ms.

```
# clia setfruledstate 20 0 0 BLINK 100 200
IPM Sentry Shelf Manager Command Line Interpreter

    Setting FRU's led state completed successfully, status = 0x0
#
```

# sethysteresis

### Syntax:

`sethysteresis` *IPMB-adresse* `[`*lun*`:]` *sensor_id* | *sensorname* `pos` | `neg` `[-r]` *wert*

### Zweck:

Dieser Befehl legt den Wert für die angegebene Hysterese des entsprechenden Sensors fest. Beim Sensor muss es sich um einen Sensor mit Schwellenwert handeln. Er muss die bezeichnete Schwellenwert-Hysterese unterstützen, und die Hysterese muss einstellbar sein.

Mithilfe dieses Befehls können Benutzer die Sensornummer mit der Nummer der logischen Einheit (Logical Unit Number, LUN) qualifizieren, wenn der Zielcontroller Sensoren auf mehreren LU-Nummern unterstützt. Dieser Befehl legt die positive Hysterese fest, wenn das Argument `pos` angegeben ist, und die negative Hysterese, wenn das Argument `neg` angegeben ist.

### Beispiele:

Festlegen der positiven Hysterese für Sensor #2 des IPM-Controllers unter der IPMB-Adresse 0xFC.

```
# clia sethysteresis FC 2 pos 10
IPM Sentry Shelf Manager Command Line Interpreter

     Positive hysteresis set successfully to 0xA, previous: 0x0
#
```

# setipmbstate

### Syntax:

`setipmbstate` *IPMB-adresse* `A|B` `[`*link*`]` `1|0` (in sternförmiger IPMB-0-Umgebung)
`setipmbstate` *IPMB-adresse* `A|B` `1|0` (in Bus-geleiteter IPMB-0-Umgebung)

### Zweck:

Dieser Befehl aktiviert oder deaktiviert einen IPMB-Link auf dem IPM-Zielcontroller. Das zweite Argument definiert, ob der Bus (IPMB-A oder IPMB-B) aktiviert oder deaktiviert wird. Das letzte Argument definiert die Operation, die ausgeführt werden soll: 1 – Link aktivieren, 0 – Link deaktivieren.

Der Befehl verhält sich für Bus-geleitete Umgebungen und sternförmige Umgebungen unterschiedlich. Wenn eine Bus-geleitete Umgebung verwendet wird oder wenn es sich beim IPM-Zielcontroller in einer sternförmigen Umgebung nicht um ein IPMB-Hub-Gerät handelt, wird das Argument *link* nicht verwendet. Bei einem IPMB-Hub-Controller in einer sternförmigen Umgebung ist das Argument *link* optional.

Wenn *link* angegeben ist, aktiviert bzw. deaktiviert der Befehl den entsprechenden sternförmigen IPMB-Link (1 bis 95). Wenn *link* nicht angegeben ist, aktiviert bzw. deaktiviert der Befehl alle Links des IPMB-Hub-Geräts im sternförmigen System.

### Beispiele:

Deaktivieren des IPMB-A-Links für den IPM-Controller unter IPMB-Adresse 92h.

```
# clia setipmbstate 92 A 0
IPM Sentry Shelf Manager Command Line Interpreter

     Command executed successfully
```

Aktivieren des sternförmigen IPMB-Links 3 (Bus B) auf dem Shelf Manager (der sich im IPMB-Hub-Gerät befindet).

```
# clia setipmbstate 20 B 3 1
IPM Sentry Shelf Manager Command Line Interpreter

     Command executed successfully
```

## setlanconfig

### Syntax:

`setlanconfig` *kanal parametername weitere_parameter*
`setlanconfig` *kanal parameternummer weitere_parameter*

### Zweck:

Dieser Befehl legt den Wert des angegebenen LAN-Konfigurationsparameters auf dem angegebenen Kanal fest. Sie müssen die Kanalnummer, den Namen bzw. die Nummer des Konfigurationsparameters und den Parameterwert explizit angeben.

In TABELLE A-3 sind Namen und Nummern von LAN-Konfigurationsparametern aufgeführt, die vom Befehl `setlanconfig` unterstützt werden:

**TABELLE A-3** LAN-Konfigurationsparameter für `setlanconfig`

| Parametername | Nummer | Beschreibung |
|---|---|---|
| `auth_enables` | 2 | Fünf 8-Bit-Werte, die Authentifizierungstypen-Aktivierungsflags für Rückmeldungs-, Benutzer-, Bediener-, Administrator- und OEM-Berechtigungsebenen für den LAN-Kanal enthalten. |
| `ip` | 3 | Ein Zeichenfolgenwert, der die IP-Adresse enthält, die dem LAN-Kanal im Dezimalnotationformat mit Punkten zugewiesen ist. |
| `subnet_mask` | 6 | Ein Zeichenfolgenwert, der die Teilnetzmaske enthält, die dem LAN-Kanal im Dezimalnotationformat mit Punkten zugewiesen ist. |
| `ipv4_hdr_param` | 7 | Drei 8-Bit-Werte, die verschiedene IPv4-Headerparameter zum Senden von RMCP-Paketen enthalten:<br>• Lebensdauer<br>• IP-Headerflags (Bits [7:5])<br>• Vorrangigkeit (Bits [7:5]) und Diensttyp (Bits [4:1]) |
| `arp_control` | 10 | Zwei Flags, die das ARP-Verhalten auf dem LAN-Kanal steuern:<br>• Aktivieren der Antworten auf ARP-Anforderungen<br>• Aktivieren des Sendens von nicht angeforderten ARPs |
| `arp_interval` | 11 | Das Intervall für nicht angeforderte ARPs im Festpunktformat (wobei der ganzzahlige Teil für die Sekunden und der Bruchteil für Millisekunden steht). |
| `dft_gw_ip` | 12 | Ein Zeichenfolgenwert, der die IP-Adresse des Standardgateways in Dezimalnotation mit Punkten enthält. |
| `backup_gw_ip` | 14 | Ein Zeichenfolgenwert, der die IP-Adresse des Sicherungsgateways in Dezimalnotation mit Punkten enthält. |

**TABELLE A-3** LAN-Konfigurationsparameter für `setlanconfig` *(Fortsetzung)*

| | | |
|---|---|---|
| `community` | 16 | Ein Zeichenfolgenwert (bis zu 18 Symbole), der bei PET-Unterbrechungen in das Feld `Community String` eingefügt wird. |
| `destination_type` | 18 | Der Zieltyp, der mithilfe der angegebenen Auswahlwerkzeugfestlegung identifiziert wird. Für diesen Parameter müssen Sie die Auswahlwerkzeugfestlegung angeben. Jeder Zieltypeintrag enthält die folgenden Felder:<br>• Zieltyp (0 bis 7)<br>• Flag für Warnmeldungsbestätigung<br>• Zeitlimit für die Warnmeldungsbestätigung/ Wiederholungsintervall in Sekunden (1 bis 256)<br>• Anzahl der Wiederholungsversuche (0 bis 7) |
| `destination_address` | 19 | Die Zieladressen, die der angegebenen Auswahlwerkzeugfestlegung zugeordnet sind. Für diesen Parameter müssen Sie die Auswahlwerkzeugfestlegung angeben. Jede Zieladresse enthält die folgenden Felder:<br>• Auswahlwerkzeugfestlegung für Gateway: 0 – Standardeinstellung verwenden, 1 – Sicherung verwenden<br>• IP-Adresse (Zeichenfolge im Dezimalformat mit Punkten)<br>• MAC-Adresse (Zeichenfolge mit sechs hexadezimalen Byte-Werten mit Doppelpunkten als Trennzeichen [Symbol :]) |

# auth_enables

### Syntax:

setlanconfig *kanal* auth_enables *wert1 wert2 wert3 wert4 wert5*
setlanconfig *kanal* 2 *wert1 wert2 wert3 wert4 wert5*

### Zweck:

Dieser Befehl legt den aktuellen Wert des LAN-Parameters `auth_enables` fest. Dieser Parameter gibt an, welche Authentifizierungstypen vom Shelf Manager für die fünf unterstützten Berechtigungsebenen momentan aktiviert sind (Rückmeldung, Benutzer, Bediener, Administrator und OEM). Die Darstellung erfolgt über fünf Bytes, die jeweils für die entsprechende Berechtigungsebene stehen und als Bitmaske behandelt werden. Die Bits sind dabei wie folgt definiert:

- 0x01 – Kein

- 0x02 – MD2
- 0x04 – MD
- 5 0x10 – Straight-Passwort/-Schlüssel
- 0x20 – OEM-abhängig

Die Parameter *wert1* bis *wert5* stehen für die Werte dieser Bytes und werden im hexadezimalen Format angegeben. Der Shelf Manager unterstützt die Berechtigungsebenen **Rückmeldung** und **OEM** momentan nicht. Aus diesem Grund müssen Sie für die Parameter *wert1* und *wert5* dieser Berechtigungsebenen den Wert 0 angeben.

#### Beispiel:

```
# clia setlanconfig 1 auth_enables 0 1 1 1 0
IPM Sentry Shelf Manager Command Line Interpreter
Authentication Type Enables set successfully
#
```

## ip

#### Syntax:

setlanconfig *kanal* ip *wert*
setlanconfig *kanal* 3 *wert*

#### Zweck:

Dieser Befehl legt die aktuelle IP-Adresse fest, die vom Kanal verwendet wird. Der Wert in Dezimalnotation mit Punkten steht für eine IP-Adresse.

#### Beispiel:

```
# clia setlanconfig 1 ip 172.16.2.203
IPM Sentry Shelf Manager Command Line Interpreter
IP Address set successfully
#
```

## subnet_mask

#### Syntax:

setlanconfig *kanal* teilnetzmaske *wert*
setlanconfig *kanal* 6 *wert*

**Zweck:**

Dieser Befehl legt die aktuelle IP-Teilnetzmaske fest, die vom Kanal verwendet wird. Der Wert in Dezimalnotation mit Punkten steht für eine Teilnetzmaske.

**Beispiel:**

```
# clia setlanconfig 1 subnet_mask 255.255.255.0
IPM Sentry Shelf Manager Command Line Interpreter
Subnet Mask set successfully
#
```

## ipv4_hdr_param

**Syntax:**

setlanconfig *kanal* ipv4_hdr_param *wert1 wert2 wert3*
setlanconfig *kanal* 7 *wert1 wert2 wert3*

**Zweck:**

Dieser Befehl legt die IP 4-Headerparameter für den Shelf Manager fest. Diese werden als drei Einzelbyte-Werte in hexadezimaler Notation angegeben: *wert1*, *wert2* und *wert3*. Der Inhalt der Bytes entspricht Abschnitt 19.2 der Spezifikation IPMI 1.5 und enthält die folgenden Attribute:

- Lebensdauer in Byte 1
- IP-Headerflags (Bits [7:5]) in Byte 2
- Vorrangigkeit (Bits [7:5]) und Diensttyp (Bits [4:1]) in Byte 3

**Beispiel:**

```
# clia setlanconfig 1 ipv4_hdr_param 37 E0 11
IPM Sentry Shelf Manager Command Line Interpreter
IPv4 Header Parameters set successfully
#
```

## arp_control

**Syntax:**

setlanconfig *kanal* arp_control *wert*
setlanconfig *kanal* 10 *wert*

### Zweck:

Dieser Befehl legt den aktuellen Wert des LAN-Parameters `arp_control` fest. Dieser Parameter gibt die zusätzliche ARP-Unterstützung an, die vom Shelf Manager bereitgestellt wird. Es wird ein einzelnes Byte verwendet, das als Bitmaske behandelt wird und bei dem die Bits wie folgt definiert sind:

- 0x01 Vom Shelf Manager erzeugte nicht angeforderte ARPs aktivieren
- 0x02 Vom Shelf Manager erzeugte ARP-Antworten aktivieren

Andere Bits sind reserviert und müssen auf 0 gesetzt sein.

### Beispiel:

```
# clia setlanconfig 1 arp_control 3
IPM Sentry Shelf Manager Command Line Interpreter
BMC-generated ARP control set successfully
#
```

## arp_interval

### Syntax:

`setlanconfig` *kanal* `arp_interval` *wert*
`setlanconfig` *kanal* `11` *wert*

### Zweck:

Dieser Befehl legt das aktuelle ARP-Intervall fest, das vom Kanal verwendet wird. Der Wert steht für die Anzahl an Sekunden/Millisekunden im numerischen Festpunktformat (ggf. mit Bruchteil). Gemäß der IPMI-Definition dieses Parameters wird der Wert auf das größte Zeitintervall beschränkt, das durch 500 Millisekunden teilbar ist.

**Beispiel:**

```
# clia setlanconfig 1 arp_interval 3.5
IPM Sentry Shelf Manager Command Line Interpreter
Gratuitous ARP interval set successfully
#
```

# dft_gw_ip

**Syntax:**

`setlanconfig` *kanal* `dft_gw_ip` *wert*
`setlanconfig` *kanal* `12` *wert*

**Zweck:**

Dieser Befehl legt die IP-Adresse des Standardgateways fest, das vom Kanal verwendet wird. Der Wert in Dezimalnotation mit Punkten steht für eine IP-Adresse.

**Beispiel:**

```
# clia setlanconfig 1 dft_gw_ip 172.16.2.100
IPM Sentry Shelf Manager Command Line Interpreter
Default Gateway Address set successfully
#
```

# backup_gw_ip

**Syntax:**

`setlanconfig` *kanal* `backup_gw_ip` *wert*
`setlanconfig` *kanal* `14` *wert*

**Zweck:**

Dieser Befehl legt die IP-Adresse des Sicherungsgateways fest, das vom Kanal verwendet wird. Der Wert in Dezimalnotation mit Punkten steht für eine IP-Adresse.

### Beispiel:

```
# clia setlanconfig 1 backup_gw_ip 172.16.2.100
IPM Sentry Shelf Manager Command Line Interpreter
Backup Gateway Address set successfully
#
```

## community

### Syntax:

`setlanconfig` *kanal* `community` *wert*
`setlanconfig` *kanal* `16` *wert*

### Zweck:

Dieser Befehl legt den Community-Zeichenfolgenparameter fest, der in PET-Unterbrechungen verwendet wird. Bei dem Wert muss es sich um eine Zeichenfolge handeln, die in doppelte Anführungszeichen gesetzt ist.

### Beispiel:

```
# clia setlanconfig 1 community "Community"
IPM Sentry Shelf Manager Command Line Interpreter
Community string set successfully
#
```

## destination_type

### Syntax:

`setlanconfig` *kanal* `destination_type` *auswahlwerkzeug_festlegen wert1 wert2 wert3*
`setlanconfig` *kanal* `18` *auswahlwerkzeug_festlegen wert1 wert2 wert3*

### Zweck:

Dieser Befehl legt das Element der Zieltabelle fest, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 0-basiert. Der Wert 0 wird für die Adressierung des flüchtigen Ziels verwendet. Die Werte *wert1*, *wert2* und *wert3* liefern gemäß Abschnitt 19.2 der IPMI-Spezifikation Informationen zum neuen Ziel. Die folgenden Informationen werden bereitgestellt:

- Typ des Warnmeldungsziels (PET-Unterbrechung oder OEM-Ziel; Angabe, ob die Warnmeldung bestätigt werden soll)
- Zeitlimit für Warnmeldungsbestätigung
- Anzahl an Wiederholungsversuchen

### Beispiel:

```
# clia setlanconfig 1 destination_type 2 80 3 5
IPM Sentry Shelf Manager Command Line Interpreter
Destination Type set successfully
#
```

## destination_address

### Syntax:

`setlanconfig` *kanal* `destination_address` *auswahlwerkzeug_festlegen gatewayauswahl IP-adresse MAC-adresse*
`setlanconfig` *kanal* `19` *auswahlwerkzeug_festlegen gatewayauswahl IP-adresse MAC-adresse*

### Zweck:

Dieser Befehl legt das Element der Zieladresstabelle fest, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 0-basiert. Der Wert 0 wird für die Adressierung des flüchtigen Ziels verwendet. Die Befehlsparameter liefern die erforderlichen Informationen:

- *gatewayauswahl* – Zu verwendendes Gateway: 0 für Standardgateway, 1 für Sicherungsgateway
- *IP-adresse* – IP-Zieladresse als Dezimalnotation mit Punkten
- *MAC-adresse* – MAC-Zieladresse, sechs durch Doppelpunkte getrennte Hexadezimal-Bytes

### Beispiel:

```
# clia setlanconfig 1 destination_address 2 0 172.16.2.100
90:93:93:93:93:93
IPM Sentry Shelf Manager Command Line Interpreter
Destination Addresses set successfully
#
```

# setlocked

### Syntax:

setlocked *IPMB-adresse fru-id* 0 |
setlocked *IPMB-adresse fru-id* 1
setlocked board *n* 0 | 1
setlocked shm *n* 0 | 1
setlocked fan_tray *n* 0 | 1

### Zweck:

Dieser Befehl setzt das Locked-Bit für die jeweilige FRU auf den angegebenen Status (0 für Entsperren und 1 für Sperren). Die FRU wird angegeben, indem die IPMB-Adresse des als Eigentümer fungierenden IPM-Controllers und die FRU-Geräte-ID verwendet wird. Die FRU-Geräte-ID 0 bezeichnet in PICMG 3.0-Kontexten den IPM-Controller.

Das Locked-Bit steuert gemäß Spezifikation PICMG 3.0, ob die FRU selbständig vom Status M1 (Inaktiv) in den Status M2 (Aktivierungsanforderung) wechseln darf. Wenn das Locked-Bit festgelegt ist, ist dieser Übergang nicht zulässig. Wenn der Shelf Manager den Deaktivierungsbefehl an die FRU sendet, geht die FRU in den Status M1 über und setzt das Locked-Bit, um nachfolgende Statusübergänge zu verhindern.

Sie können diesen Befehl verwenden, um eine zuvor manuell deaktivierte FRU wieder zu aktivieren, indem Sie das entsprechende Locked-Bit löschen.

### Beispiel:

Richtiges Löschen des Locked-Bits für den IPM-Controller unter Adresse 9C, um ihn wieder zu aktivieren.

```
# clia setlocked 9c 0 0
IPM Sentry Shelf Manager Command Line Interpreter
Lock set successfully to 0x0
#
```

# setpefconfig

### Syntax:

setpefconfig *parametername weitere_parameter*
setpefconfig *parameternummer weitere_parameter*

### Zweck:

Dieser Befehl legt für den angegebenen PEF-Konfigurationsparameter einen neuen Wert fest. In TABELLE A-4 sind Namen und Nummern von PEF-Konfigurationsparametern aufgeführt, die Sie mithilfe dieses Befehls festlegen können.

**TABELLE A-4** PEF-Konfigurationsparameter für `setpefconf`

| Parametername | Nummer | Beschreibung |
|---|---|---|
| `control` | 1 | Ein 8-Bit-Wert, der für PEF-Steuerflags steht (PEF aktivieren, PEF-Startverzögerung aktivieren usw.). |
| `action_control` | 2 | Ein 8-Bit-Wert, der für globale Steuerflags für PEF-Aktionen steht (Zurücksetzen aktivieren, Herunterfahren aktivieren usw.). |
| `startup_delay` | 3 | Zeitraum für die PEF-Verzögerung nach dem Hochfahren bzw. Zurücksetzen des Systems in Sekunden. |
| `alert_startup_delay` | 4 | Zeitraum für die Verzögerungswarnmeldungen nach dem Hochfahren bzw. Zurücksetzen des Systems in Sekunden. |

**TABELLE A-4** PEF-Konfigurationsparameter für `setpefconf` *(Fortsetzung)*

| | | |
|---|---|---|
| `event_filter` | 6 | Ein Ereignisfilter-Tabelleneintrag, für den die angegebene Auswahlwerkzeugfestlegung steht. Es gelten die folgenden 19 numerischen hexadezimalen Werte, die gemäß der Definition in Tabelle 15-2 der IPMI-Spezifikation, Version 1.5, codiert sind:<br>• Filterkonfiguration<br>• Ereignisfilteraktion<br>• Nummer des Warnverfahrens<br>• Ereignisschweregrad<br>• Generator-ID, Byte 1<br>• Generator-ID, Byte 2<br>• Sensortyp<br>• Sensornummer<br>• Ereignisauslöser (Ereignis-/Lesetyp)<br>• Ereignisdaten 1, Ereignisversatzmaske<br>• Ereignisdaten 1, AND-Maske<br>• Ereignisdaten 1, Compare 1<br>• Ereignisdaten 1, Compare 2<br>• Ereignisdaten 2, AND-Maske<br>• Ereignisdaten 2, Compare 1<br>• Ereignisdaten 2, Compare 2<br>• Ereignisdaten 3, AND-Maske<br>• Ereignisdaten 3, Compare 1<br>• Ereignisdaten 3, Compare 2 |
| `event_filter_data1` | 7 | Das erste Byte des Ereignisfilter-Tabelleneintrags, für den die angegebene Auswahlwerkzeugfestlegung steht. |
| `alert_policy` | 9 | Ein Warnverfahren-Tabelleneintrag, für den die angegebene Auswahlwerkzeugfestlegung steht. Es gelten die folgenden fünf numerischen hexadezimalen Werte, die gemäß der Definition in Tabelle 15-4 der IPMI-Spezifikation, Version 1.5, codiert sind:<br>• Verfahrensnummer (4-Bit-Wert)<br>• Verfahren (4-Bit-Wert); enthält das Bit zum Aktivieren bzw. Deaktivieren<br>• Kanalnummer (4-Bit-Wert)<br>• Zielauswahlwerkzeug (4-Bit-Wert)<br>• Auswahlwerkzeugfestlegung für Warnmeldungszeichenfolge |
| `system-guid` | 10 | Eine GUID, die bei der PET-Unterbrechung in das GUID-Feld eingefügt wird. |

**TABELLE A-4** PEF-Konfigurationsparameter für `setpefconf` *(Fortsetzung)*

| | | |
|---|---|---|
| `alert_string_key` | 12 | Ein Warnmeldungszeichenfolgen-Schlüssel, für den die angegebene Auswahlwerkzeugfestlegung steht. Er besteht aus zwei 8-Bit-Werten: Ereignisfilternummer und Warnmeldungszeichenfolgen-Einstellung. |
| `alert_string` | 13 | Eine Warnmeldungszeichenfolge, für die die angegebene Auswahlwerkzeugfestlegung steht. |
| `oem_filter` | 97 | Ein OEM-Filter-Tabelleneintrag, für den die angegebene Auswahlwerkzeugfestlegung steht. Er umfasst die folgenden drei numerischen Werte:<br>• Byte 1: Untergrenze für SEL-Datensatztypbereich<br>• Byte 2: Obergrenze für SEL-Datensatztypbereich<br>• Byte 3: Warnverfahrennummer, die für SEL-Einträge aufgerufen wird, deren Datensatztypen dem oben angegebenen Bereich entsprechen. |

# control

### Syntax:

```
setpefconfig control wert
setpefconfig 1 wert
```

### Zweck:

Dieser Befehl legt einen neuen Wert für den PEF-Parameter `control` fest. Für den Parameter wird ein einzelnes Byte verwendet, das als Bitmaske behandelt wird und bei dem die Bits wie folgt definiert sind:

- 0x01 – PEF aktivieren
- 0x02 – Erzeugen von Ereignismeldungen für PEF-Aktionen aktivieren
- 0x04 – PEF-Startverzögerungen beim Hochfahren und Zurücksetzen des Systems aktivieren
- 0x08 – Startverzögerungen für PEF-Warnmeldungen aktivieren

Andere Bits sind reserviert und müssen auf 0 gesetzt sein. Außerdem müssen Sie den Wert in hexadezimalem Format angeben.

### Beispiel:

```
# clia setpefconfig control 7
IPM Sentry Shelf Manager Command Line Interpreter
PEF control set successfully
#
```

## action_control

### Syntax:

`setpefconfig action_control` *wert*
`setpefconfig 2` *wert*

### Zweck:

Dieser Befehl legt einen neuen Wert für den PEF-Parameter `action_control` fest. Für den Parameter wird ein einzelnes Byte verwendet, das als Bitmaske behandelt wird und bei dem die Bits wie folgt definiert sind:

- 0x01 – Warnmeldungsaktion aktivieren
- 0x02 – Aktion zum Herunterfahren aktivieren
- 0x04 – Aktion zum Zurücksetzen aktivieren
- 0x08 – Aktion für Neustart (Power Cycle) aktivieren
- 0x10 – OEM-Aktion aktivieren
- 0x20 – Diagnoseunterbrechung aktivieren

Andere Bits sind reserviert und müssen auf 0 gesetzt sein. Außerdem müssen Sie den Wert in hexadezimalem Format angeben.

### Beispiel:

```
# clia setpefconfig action_control 3f
IPM Sentry Shelf Manager Command Line Interpreter
PEF action control set successfully
#
```

## startup_delay

### Syntax:

`setpefconfig startup_delay` *wert*
`setpefconfig 3` *wert*

### Zweck:

Dieser Befehl legt den neuen Wert des PEF-Parameters `startup_delay` fest. Bei diesem Parameter handelt es sich um ein einzelnes Byte, das für die Anzahl an Sekunden steht, um die das PEF-Dienstprogramm beim Starten verzögert wird. Der Wert wird als Dezimalzahl in Sekunden angegeben.

### Beispiel:

```
# clia setpefconfig startup_delay 45
IPM Sentry Shelf Manager Command Line Interpreter
PEF startup delay set successfully
#
```

## alert_startup_delay

### Syntax:

`setpefconfig startup_delay` *wert*
`setpefconfig 4` *wert*

### Zweck:

Dieser Befehl legt den aktuellen Wert des PEF-Parameters `alert_startup_delay` fest. Bei diesem Parameter handelt es sich um ein einzelnes Byte, das für die Anzahl an Sekunden steht, um die das Warnmeldungsdienstprogramm beim Starten verzögert wird. Der Wert wird als Dezimalzahl in Sekunden angegeben.

### Beispiel:

```
# clia setpefconfig alert_startup_delay 45
IPM Sentry Shelf Manager Command Line Interpreter
Alert startup delay set successfully
#
```

## event_filter

### Syntax:

`setpefconfig event_filter` *auswahlwerkzeug_festlegen wert1* … `<wert19>`
`setpefconfig 6` *auswahlwerkzeug_festlegen wert1* … `<wert19>`

### Zweck:

Dieser Befehl legt das Element der Ereignisfiltertabelle fest, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 1-basiert. Der Inhalt des neuen Elements wird mithilfe von 19 numerischen hexadezimalen Werten (*wert1* bis <wert19>) angegeben, die gemäß der Definition in Tabelle 15-2 der IPMI-Spezifikation, Version 1.5, codiert sind:

- Filterkonfiguration
- Ereignisfilteraktion
- Nummer des Warnverfahrens
- Ereignisschweregrad
- Generator-ID, Byte 1
- Generator-ID, Byte 2
- Sensortyp
- Sensornummer
- Ereignisauslöser (Ereignis-/Lesetyp)
- Ereignisdaten 1, Ereignisversatzmaske
- Ereignisdaten 1, AND-Maske
- Ereignisdaten 1, Compare 1
- Ereignisdaten 1, Compare 2
- Ereignisdaten 2, AND-Maske
- Ereignisdaten 2, Compare 1
- Ereignisdaten 2, Compare 2
- Ereignisdaten 3, AND-Maske
- Ereignisdaten 3, Compare 1
- Ereignisdaten 3, Compare 2

### Beispiel:

Festlegen eines Ereignisfilters 2, um eine Warnmeldungsaktion auszulösen, wenn ein IPM-Controller unter Adresse 9C, FRU 0, den Status M0 erreicht (die Warnmeldung wird gemäß Warnmeldungsverfahren #1 gesendet):

```
# clia setpefconfig event_filter 2 80 1 1 10 9C FF F0 FF FF FF FF
0F FF 0 0 0 0 FF FF 0
IPM Sentry Shelf Manager Command Line Interpreter
Event filter set successfully
#
```

# event_filter_data1

### Syntax:

setpefconfig event_filter_data1 *auswahlwerkzeug_festlegen wert*
setpefconfig 7 *auswahlwerkzeug_festlegen wert*

### Zweck:

Dieser Befehl legt das erste Byte des Elements der Ereignisfiltertabelle fest, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 1-basiert. Dieses Byte wird im hexadezimalen Format angegeben. Die Bits haben dabei die folgende Bedeutung:

- 0x80 – Dieser Filter ist aktiviert.
- 0x40 – Dieser Filter ist vom Hersteller vorkonfiguriert und darf mithilfe der Software nicht geändert werden.

Andere Bits sind reserviert und müssen auf 0 gesetzt sein.

Sie können diesen Befehl verwenden, um den Status „Aktiviert“ bzw. „Deaktiviert“ für einen Ereignisfilter schnell umzuschalten. Sie können diesen Schritt also ausführen, ohne dass Sie den gesamten Tabelleneintrag neu schreiben müssen.

### Beispiele:

Aktivieren von Ereignisfilter 2.

```
# clia setpefconfig event_filter_data1 2 80
IPM Sentry Shelf Manager Command Line Interpreter
Event filter data1 set successfully
#
```

Deaktivieren von Ereignisfilter 2.

```
# clia setpefconfig event_filter_data1 2 0
IPM Sentry Shelf Manager Command Line Interpreter
Event filter data1 set successfully
#
```

# alert_policy

### Syntax:

`setpefconfig alert_policy` *auswahlwerkzeug_festlegen wert1 wert2 wert3 wert4 wert5*
`setpefconfig 9` *auswahlwerkzeug_festlegen wert1 wert2 wert3 wert4 wert5*

### Zweck:

Dieser Befehl legt einen Warnverfahren-Tabelleneintrag fest, für den die angegebene Auswahlwerkzeugfestlegung steht. Der Inhalt des neuen Elements wird mithilfe der folgenden fünf numerischen hexadezimalen Werte (*wert1* bis *wert5*) angegeben, die gemäß der Definition in Tabelle 15-4 der IPMI-Spezifikation, Version 1.5, codiert sind:

- Verfahrensnummer (4-Bit-Wert)
- Verfahren (4-Bit-Wert); enthält das Bit zum Aktivieren bzw. Deaktivieren
- Kanalnummer (4-Bit-Wert)
- Zielauswahlwerkzeug (4-Bit-Wert)
- Auswahlwerkzeugfestlegung für Warnmeldungszeichenfolge

### Beispiel:

Im folgenden Beispiel wird der Warnverfahren-Tabelleneintrag 2 mit den folgenden Attributen eingerichtet:

- Nummer des Verfahrens = 5
- Aktiviert
- Verfahren = Warnmeldung immer an dieses Ziel senden
- Zielkanal = 1
- Zielauswahlwerkzeug = 1
- Auswahl der Warnmeldungszeichenfolge = Für alle Ereignisse Zeichenfolge 1 verwenden

```
# clia setpefconfig alert_policy 2 5 8 1 1 1
IPM Sentry Shelf Manager Command Line Interpreter
Policy set successfully
#
```

## system_guid

### Syntax:

setpefconfig system_guid *guid-wert*
setpefconfig 10 *guid-wert*
setpefconfig system_guid none
setpefconfig 10 none

### Zweck:

Dieser Befehl legt den aktuellen Wert des PEF-Parameters `system_guid` fest. Dieser Parameter steht für die GUID, die in einer PET-Unterbrechungs-PDU an ein Warnmeldungsziel gesendet wird. Sie können diese GUID als separate GUID oder identisch mit der System-GUID definieren.

Sie können als *guid-wert* eine tatsächliche GUID angeben, die dem GUID-Standardformat `xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx` entspricht, oder Sie können einen symbolischen Wert `none` verwenden. Im ersten Fall verwendet das PEF-Dienstprogramm die angegebene GUID in PET-Unterbrechungen. Im zweiten Fall verwendet das PEF-Dienstprogramm standardmäßig die System-GUID (Ergebnis des IPMI-Befehls `Get System GUID`) für PET-Unterbrechungen.

### Beispiele:

```
# clia setpefconfig system_guid 23662F7F-BA1B-4b65-8808-
94CA09C9BBB0
IPM Sentry Shelf Manager Command Line Interpreter
GUID set successfully
#
# clia setpefconfig system_guid none
IPM Sentry Shelf Manager Command Line Interpreter
Using the system GUID
#
```

## alert_string_key

### Syntax:

setpefconfig alert_string_key *auswahlwerkzeug_festlegen wert1 wert2*
setpefconfig 12 *auswahlwerkzeug_festlegen wert1 wert2*

### Zweck:

Dieser Befehl legt das Element der Warnmeldungszeichenfolgen-Schlüsseltabelle fest, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 1-basiert. Sie können als Auswahlwerkzeugfestlegung einen Wert von 0 verwenden, um die flüchtige Warnmeldungszeichenfolge anzugeben. Jeder Schlüssel weist einem Ereignisfilter eine Warnmeldungszeichenfolge zu, damit Warnmeldungen erzeugt werden können. Er besteht aus der Ereignisfilternummer und der Nummer der Warnmeldungszeichenfolge. Bei beiden Werten handelt es sich um 8-Bit-Werte, die mithilfe der Parameter *wert1* und *wert2* im hexadezimalen Format angegeben werden.

### Beispiel:

```
# clia setpefconfig alert_string_key 2 10 11
IPM Sentry Shelf Manager Command Line Interpreter
Alert string keys set successfully
#
```

## alert_string

### Syntax:

setpefconfig alert_string *auswahlwerkzeug_festlegen* <zeichenfolgenwert>
setpefconfig 13 *auswahlwerkzeug_festlegen* <zeichenfolgenwert>

### Zweck:

Dieser Befehl legt das Element der Warnmeldungszeichenfolgen-Tabelle an, wobei der Index *auswahlwerkzeug_festlegen* entspricht. Indizes sind 1-basiert. Sie können einen Indexwert von 0 verwenden, um die flüchtige Warnmeldungszeichenfolge anzugeben. Der Zeichenfolgenwert muss in doppelte Anführungszeichen (") gesetzt werden und kann innerhalb der Anführungszeichen Sonderzeichen und Zeilenschaltungen enthalten.

### Beispiel:

```
# clia setpefconfig alert_string 2 "This string has a line feed
inside."
IPM Sentry Shelf Manager Command Line Interpreter
Alert string set successfully
#
```

## oem_filter

### Syntax:

`setpefconfig oem_filter` *auswahlwerkzeug_festlegen wert1 wert2 wert3*
`setpefconfig 97` *auswahlwerkzeug_festlegen wert1 wert2 wert3*

### Zweck:

Die OEM-Filtertabelle ist eine Pigeon Point Systems-definierte OEM-Erweiterung der IPMI-Spezifikation. Sie ermöglicht es, dass PEF-Parameter zusätzlich zu Plattformereignissen auf SEL-Einträge mit OEM-Zeitstempel und ohne Zeitstempel (Datensatztypbereich C0h bis FFh) angewendet werden.

Jeder Eintrag der OEM-Filtertabelle definiert den Bereich der Datensatztypen (im Bereich der OEM-Datensatztypen), für den der jeweilige OEM-Filter gilt. Außerdem definiert er die Warnverfahrennummer, die aufgerufen wird, wenn ein Datensatz mit übereinstimmendem Datensatztyp in das SEL eingefügt wird.

Dieser Befehl legt einen OEM-Filter-Tabelleneintrag fest, für dessen Nummer der angegebene Wert der Auswahlwerkzeugfestlegung steht. Der Eintrag umfasst die folgenden drei numerischen Werte:

- Byte 1: Untergrenze für SEL-Datensatztypbereich
- Byte 2: Obergrenze für SEL-Datensatztypbereich
- Byte 3: Warnverfahrennummer, die für SEL-Einträge aufgerufen wird, deren Datensatztypen dem oben angegebenen Bereich entsprechen.

**Beispiel:**

```
# clia getpefconfig oem_filter
IPM Sentry Shelf Manager Command Line Interpreter
Active OEM Filters:
    0x01: OEM range boundary 0xff:0xff, alert policy # 1
#
# clia setpefconfig oem_filter 4 0xdc 0xf3 5
IPM Sentry Shelf Manager Command Line Interpreter
OEM filter set successfully
#
# clia getpefconfig oem_filter
IPM Sentry Shelf Manager Command Line Interpreter
Active OEM Filters:
    0x01: OEM range boundary 0xff:0xff, alert policy # 1
    0x04: OEM range boundary 0xdc:0xf3, alert policy # 5
#
```

# setsensoreventenable

**Syntax:**

`setsensoreventenable` *IPMB-adresse sensorname* `global [assertion_events [deassertion_events]]`

`setsensoreventenable` *IPMB-adresse* `[`*lun*`:]`*sensornummer* `global [assertion_events [deassertion_events]]`

Anstelle der *IPMB-adresse* kann der Benutzer Folgendes verwenden:

`board` *n*
`shm` *n*

**Zweck:**

Dieser Befehl ändert die Ereignisaktivierungsmaske für den angegebenen Sensor. Der Sensor wird mithilfe der IPMB-Adresse des als Eigentümer fungierenden IPM-Controllers und des Sensornamens bzw. der Sensornummer angegeben. Alternativ dazu können Sie auch die Board-Nummer oder die Nummer des dedizierten Shelf Managers verwenden, um den IPM-Zielcontroller zu bezeichnen.

Mithilfe dieses Befehls können Benutzer die Sensornummer mit der Nummer der logischen Einheit (Logical Unit Number, LUN) qualifizieren, wenn der Zielcontroller Sensoren auf mehreren LU-Nummern unterstützt. *lun* kann den Wert 0, 1 oder 3

haben. (LUN 2 ist reserviert.) Wenn die LU-Nummer fehlt, wird der Befehl auf den Sensor mit der angegebenen Sensornummer und der niedrigsten LU-Nummer angewendet. (Wenn der Befehl z. B. Sensor 3 ohne ausdrückliche LUN-Qualifizierung angibt und der Zielcontroller Sensor 3 unter LUN 1 und einen anderen Sensor 3 unter LUN 3 offenlegt, wird der Befehl auf den Sensor 3 unter LUN 1 angewendet.)

**Beispiele:**

Aktivieren des Ereignisses `Lower Non-Critical Going Low` für den Temperatursensor `Local Temp` auf dem IPM-Controller FE.

```
# clia setsensoreventenable fe "Local Temp" 0x90 0x01 0x00
IPM Sentry Shelf Manager Command Line Interpreter
Event enable mask set successfully
#
# clia getsensoreventenable -v fe "Local Temp"
IPM Sentry Shelf Manager Command Line Interpreter
fe: LUN: 0, Sensor # 3 ("Local Temp")
    Type: Threshold (0x01), "Temperature" (0x01)
    Sensor scanning disabled
    Assertion event mask:   0x0001
        Assertion event for "Lower Non-Critical Going Low" enabled
    Deassertion event mask: 0x0000
#
```

Durchführen der gleichen Operation für den Sensor, wobei der Sensor jedoch mithilfe der LU-Nummer und der Sensornummer angegeben wird:

```
# clia setsensoreventenable fe 0:3 0x90 0x01 0x00
IPM Sentry Shelf Manager Command Line Interpreter
Event enable mask set successfully
#
```

# setthreshold

### Syntax:

`setthreshold` *IPMB-adresse sensorname schwellenwerttyp* `[-r]` *wert*
`setthreshold` *IPMB-adresse* `[`*lun*`:]`*sensornummer schwellenwerttyp* `[-r]` *wert*

Anstelle der *IPMB-adresse* können Benutzer Folgendes verwenden:

`board` *n*
`shm` *n*

### Zweck:

Dieser Befehl ändert den aktuellen Schwellenwert für die angegebene Schwelle des angegebenen Sensors. Der Sensor wird mithilfe der IPMB-Adresse des als Eigentümer fungierenden IPM-Controllers und des Sensornamens bzw. der Sensornummer angegeben. Beim Zielsensor muss es sich um einen Sensor mit Schwellenwert handeln. Sie können den Parameter *schwellenwerttyp* als einen der folgenden symbolischen Werte angeben:

- `upper_non_recoverable` (kann als `unr` abgekürzt werden)
- `upper_critical` (kann als `uc` abgekürzt werden)
- `upper_non_critical` (kann als `unc` abgekürzt werden)
- `lower_non_recoverable` (kann als `lnr` abgekürzt werden)
- `lower_critical` (kann als `lc` abgekürzt werden)
- `lower_non_critical` (kann als `lnc` abgekürzt werden)

Standardmäßig wird der Zielwert in verarbeiteter Form angegeben (also in Volt für Voltsensoren oder in Celsius für Temperatursensoren). Die Option `-r` bedeutet, dass stattdessen ein Rohwert verwendet wird (normalerweise eine Byte-Größenangabe, die gemäß sensorspezifischen Regeln konvertiert wird).

Mithilfe dieses Befehls können Benutzer die Sensornummer mit der Nummer der logischen Einheit (Logical Unit Number, LUN) qualifizieren, wenn der Zielcontroller Sensoren auf mehreren LU-Nummern unterstützt. *lun* kann den Wert 0, 1 oder 3 haben. (LUN 2 ist reserviert.) Wenn die LU-Nummer fehlt, wird der Befehl auf den Sensor mit der angegebenen Sensornummer und der niedrigsten LU-Nummer angewendet. (Wenn der Befehl z. B. Sensor 3 ohne ausdrückliche LUN-Qualifizierung angibt und der Zielcontroller Sensor 3 unter LUN 1 und einen anderen Sensor 3 unter LUN 3 offenlegt, wird der Befehl auf den Sensor 3 unter LUN 1 angewendet.)

**Beispiel:**

Festlegen des oberen nicht-kritischen Schwellenwerts für den Temperatursensor `emulated temp` des IPM-Controllers 9C auf 99 Grad Celsius.

```
# clia threshold 9c 2
IPM Sentry Shelf Manager Command Line Interpreter
9c: LUN: 0, Sensor # 2 ("emulated temp")
    Type: Threshold (0x01), "Temperature" (0x01)
        Lower Non-Critical Threshold, Raw Data: 0x03, Processed
Data: 3.000000 degrees C
       Lower Critical Threshold, Raw Data: 0x14, Processed Data:
20.000000 degrees C
       Lower Non-Recoverable Threshold, Raw Data: 0xfb, Processed
Data: -5.000000 degrees C
        Upper Non-Critical Threshold, Raw Data: 0x46, Processed
Data: 70.000000 degrees C
       Upper Critical Threshold, Raw Data: 0x50, Processed Data:
80.000000 degrees C
       Upper Non-Recoverable Threshold, Raw Data: 0x5a, Processed
Data: 90.000000 degrees C
#
# clia setthreshold 9c 0:2 unc 99
IPM Sentry Shelf Manager Command Line Interpreter
Threshold set successfully
#
# clia threshold 9c 0:2
IPM Sentry Shelf Manager Command Line Interpreter
9c: LUN: 0, Sensor # 2 ("emulated temp")
    Type: Threshold (0x01), "Temperature" (0x01)
        Lower Non-Critical Threshold, Raw Data: 0x03, Processed
Data: 3.000000 degrees C
       Lower Critical Threshold, Raw Data: 0x14, Processed Data:
20.000000 degrees C
       Lower Non-Recoverable Threshold, Raw Data: 0xfb, Processed
Data: -5.000000 degrees C
        Upper Non-Critical Threshold, Raw Data: 0x63, Processed
Data: 99.000000 degrees C
       Upper Critical Threshold, Raw Data: 0x50, Processed Data:
80.000000 degrees C
       Upper Non-Recoverable Threshold, Raw Data: 0x5a, Processed
Data: 90.000000 degrees C
#
```

# shelf

### Syntax:

`shelf` *Unterbefehl*

Die folgenden Unterbefehle werden unterstützt.

- `address_table`
- `cooling_state`
- `fans_state`
- `power_distribution`
- `power_management`
- `pci_connectivity`
- `ha_connectivity`
- `h110_connectivity`
- `point-to-point_connectivity`
- `MaxCurrent [feed]` *Ampere*
- `MinVoltage [feed]` *Volt*
- `Activation` *adresse fru_id* `1|0`
- `Deactivation` *adresse fru_id* `1|0`
- `PwrCapability` *adresse fru_id Watt*
- `PwrDelay` *adresse fru_id zehntelsekunden*
- `Allowance` *sekunden*
- `PwrReorder` *adresse1 fru_id1* `before|after` *adresse2 fru_id2*
- `info_refresh`
- `info_force_update`

### Zweck:

Der Befehl `shelf` zeigt wichtige Shelf-FRU-Informationen sowie ausgewählte aktuelle Betriebsdaten für das Shelf an und ermöglicht es, einige Felder in den Shelf-FRU-Informationen zu ändern. Die Art von Informationen, die dieser Befehl anzeigt oder ändert, ist im Befehlsparameter angegeben.

In den folgenden Unterabschnitten wird die Syntax des Befehls `shelf` für verschiedene Anwendungsmöglichkeiten beschrieben.

# Anzeigen von Shelf-FRU-Informationen

### Syntax:

```
shelf [cooling_state | fans_state | address_table
       | power_distribution | power_management
       | pci_connectivity | ha_connectivity
       | h110_connectivity | point-to-point_connectivity]
```

### Zweck:

Mithilfe dieser Syntax des Befehls `shelf` zeigen Sie wichtige Shelf-FRU-Informationen und ausgewählte aktuelle Betriebsdaten für das Shelf an. Die Art von Informationen, die dieser Befehl anzeigt, ist im Befehlsparameter angegeben. In TABELLE A-5 sind die Parameter aufgeführt, die der Befehl `shelf` unterstützt:

**TABELLE A-5** Parameter des Befehls `shelf`

| Befehlsparameter | Bereitgestellte Informationen |
|---|---|
| `cooling_state`<br>(kann als `cs` abgekürzt werden) | Zeigt den aktuellen Lüftungsstatus des Shelfs an:<br>• Normal – Alle Temperatursensoren zeigen normale Betriebstemperatur an.<br>• Minor Alert – Mindestens ein Temperatursensor befindet sich im Warnmeldungsstatus **Minor**. Keiner der Sensoren befindet sich im Warnmeldungsstatus **Major** oder **Critical**.<br>• Major Alert – Mindestens ein Temperatursensor befindet sich im Warnmeldungsstatus **Major**. Keiner der Sensoren befindet sich im Warnmeldungsstatus **Critical**.<br>• Critical Alert – Mindestens ein Temperatursensor befindet sich im Warnmeldungsstatus **Critical**. |
| `fans_state`<br>(kann als `fs` abgekürzt werden) | Zeigt den aktuellen Status des Lüftertachometers im Shelf an:<br>• Normal – Alle Lüftertachometersensoren zeigen normale Betriebsgeschwindigkeit an.<br>• Minor Alert – Mindestens ein Lüftertachometersensor befindet sich im Warnmeldungsstatus **Minor**. Keiner der Sensoren befindet sich im Warnmeldungsstatus **Major** oder **Critical**.<br>• Major Alert – Mindestens ein Lüftertachometersensor befindet sich im Warnmeldungsstatus **Major**. Keiner der Sensoren befindet sich im Warnmeldungsstatus **Critical**.<br>• Critical Alert – Mindestens ein Lüftertachometersensor befindet sich im Warnmeldungsstatus **Critical**. |

**TABELLE A-5** Parameter des Befehls `shelf` *(Fortsetzung)*

| | |
|---|---|
| `address_table`<br>(kann als `at` abgekürzt werden) | Zeigt den Adresstabellendatensatz in den Shelf-FRU-Informationen an. Die folgenden Informationen werden bereitgestellt:<br>• Shelf-Adresse (je nach Typ)<br>• Liste der Adresstabelleneinträge jeweils mit Hardwareadresse, Standortnummer und Standorttyp |
| `power_distribution`<br>(kann als `pd` abgekürzt werden) | Die folgenden Informationen werden für die einzelnen Stromzuleitungen (Feeds) bereitgestellt (hauptsächlich aus dem Shelf-Stromverteilungs-Datensatz der Shelf-FRU-Informationen (Shelf Power Distribution)):<br>• Maximum External Available Current (Maximale extern verfügbare Strommenge)<br>• Maximum Internal Current (Maximale interne Strommenge)<br>• Minimum Expected Operating Voltage (Minimale erwartete Betriebsspannung)<br>• Actual Power Available (Verfügbare Ist-Leistung)<br>• Currently Used Power (Momentan verwendete Leistung)<br>• Liste der FRUs, die an die Zuleitung angeschlossen sind, jeweils mit Hardwareadresse und FRU-Geräte-ID |
| `power_management`<br>(kann als `pm` abgekürzt werden) | • Der Shelf-Energieverwaltungs-Datensatz (Shelf Power Management) aus den Shelf-FRU-Informationen. Dieser Datensatz enthält eine Liste der FRU-Leistungsdeskriptoren. Für jeden Deskriptor werden die folgenden Informationen bereitgestellt:<br>• Hardware Address (Hardwareadresse)<br>• FRU Device ID (FRU-Geräte-ID)<br>• Maximum FRU Power Capability (Maximale FRU-Leistungskapazität)<br>• Shelf Manager Controlled Activation (Vom Shelf Manager gesteuerte Aktivierung)<br>• Delay Before Next Power On (Verzögerung vor nächstem Einschaltvorgang) |

**TABELLE A-5** Parameter des Befehls `shelf` *(Fortsetzung)*

| | |
|---|---|
| `pci_connectivity`<br>(kann als `pcic` abgekürzt werden) | Der Shelf-PCI-Konnektivitäts-Datensatz (Shelf PCI Connectivity) aus den Shelf-FRU-Informationen. Die folgenden Informationen werden bereitgestellt:<br>• Deskriptor für PCI-Steckplatz<br>• DSEL-Verbindung<br>• Segment-ID<br>• Deskriptor für erweiterten PCI-Steckplatz<br>• Geografische Adresse<br>• Schnittstellennummer<br>• Systemsteckplatz-Fähigkeit |
| `ha_connectivity`<br>(kann als `ha` abgekürzt werden) | Der Shelf-HA-Konnektivitäts-Datensatz (Shelf HA Connectivity) aus den Shelf-FRU-Informationen. Die folgenden Informationen werden bereitgestellt:<br>• Unterstützung der Konnektivität für sternförmigen Betrieb |
| `h110_connectivity`<br>(kann als `h110c` abgekürzt werden) | Der Shelf-H110-Konnektivitäts-Datensatz (Shelf H110 Connectivity) aus den Shelf-FRU-Informationen. Die folgenden Informationen werden bereitgestellt:<br>• Geografische Adresse<br>• Segment-ID |
| `point-to-point_connectivity`<br>(kann als `ppc` abgekürzt werden) | Der Shelf-Point-to-Point-Konnektivitäts-Datensatz (Shelf Point-to-Point Connectivity) aus den Shelf-FRU-Informationen. Die folgenden Informationen werden bereitgestellt:<br>• Kanaltyp<br>• Kanalanzahl<br>• Steckplatz/Hardwareadresse<br>• Kanaldeskriptor |

Für die Befehlsparameter `cooling_state` und `fans_state` ist die Option `-v` für den ausführlichen Modus verfügbar. Geben Sie diese Option vor dem folgenden Befehlsparameter ein: `clia shelf -v cooling_state`. Wenn er verwendet wird, zeigt der Befehl die Liste der Sensoren an (Temperatur oder Lüftertachometer), die zum aktuellen Status beitragen. Jeder Sensor wird als Tupel angezeigt (*IPMB-adresse*, sensornummer).

### Beispiele:

Abrufen des Shelf-Lüftungsstatus.

```
# clia shelf cooling_state
IPM Sentry Shelf Manager Command Line Interpreter
Cooling state: "Normal"
#
```

Abrufen des Status für das Shelf-Lüftertachometer (ausführlich).

```
# clia shelf -v fans_state
IPM Sentry Shelf Manager Command Line Interpreter
Fans state: "Major Alert"
 Sensor(s) at this state: (0x7e,10) (0x7e,11) (0x7e,12) (0x7e,13)
                          (0x7e,14) (0x7e,15) (0x7e,16) (0x7e,17)
#
```

Abrufen der Adresstabelle.

```
# clia shelf address_table
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Address Table Record (ID=0x10)
        Version = 1
    Shelf Address            =
    Address Table Entries#   = 16
    Hw Addr: 41, Site # 1, Type: "AdvancedTCA Board" 00
    Hw Addr: 42, Site # 2, Type: "AdvancedTCA Board" 00
    Hw Addr: 43, Site # 3, Type: "AdvancedTCA Board" 00
    Hw Addr: 44, Site # 4, Type: "AdvancedTCA Board" 00
    Hw Addr: 45, Site # 5, Type: "AdvancedTCA Board" 00
    Hw Addr: 46, Site # 6, Type: "AdvancedTCA Board" 00
    Hw Addr: 47, Site # 7, Type: "AdvancedTCA Board" 00
    Hw Addr: 48, Site # 8, Type: "AdvancedTCA Board" 00
    Hw Addr: 49, Site # 9, Type: "AdvancedTCA Board" 00
    Hw Addr: 4a, Site # 10, Type: "AdvancedTCA Board" 00
    Hw Addr: 4b, Site # 11, Type: "AdvancedTCA Board" 00
    Hw Addr: 4c, Site # 12, Type: "AdvancedTCA Board" 00
    Hw Addr: 4d, Site # 13, Type: "AdvancedTCA Board" 00
    Hw Addr: 4e, Site # 14, Type: "AdvancedTCA Board" 00
#
```

Abrufen der Stromverteilungsinformationen.

```
# clia shelf power_distribution
IPM Sentry Shelf Manager Command Line Interpreter
Power Distribution:
    Feed count: 1
    Feed 00:
        Maximum External Available Current: 50.0 Amps
        Maximum Internal Current: Not specified
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 2025.000 Watts
        Currently Used Power: 160.000 Watts
        Feed-to-FRU Mapping entries count: 16
           FRU Addr: 41, FRU ID: fe
           FRU Addr: 42, FRU ID: fe
           FRU Addr: 43, FRU ID: fe
           FRU Addr: 44, FRU ID: fe
           FRU Addr: 45, FRU ID: fe
           FRU Addr: 46, FRU ID: fe
           FRU Addr: 47, FRU ID: fe
           FRU Addr: 48, FRU ID: fe
           FRU Addr: 49, FRU ID: fe
           FRU Addr: 4a, FRU ID: fe
           FRU Addr: 4b, FRU ID: fe
           FRU Addr: 4c, FRU ID: fe
           FRU Addr: 4d, FRU ID: fe
           FRU Addr: 4e, FRU ID: fe
           FRU Addr: 4f, FRU ID: fe
           FRU Addr: 50, FRU ID: fe
#
```

Abrufen der Stromverwaltungsinformationen.

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
        Version = 0
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 16
    Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds
    Hw Address: 42, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
```

```
          Shelf Manager Controlled Activation: Enabled
          Delay Before Next Power On: 0.0 seconds
     Hw Address: 43, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
          Shelf Manager Controlled Activation: Enabled
          Delay Before Next Power On: 0.0 seconds
     Hw Address: 44, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
          Shelf Manager Controlled Activation: Enabled
          Delay Before Next Power On: 0.0 seconds
     Hw Address: 45, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
          Shelf Manager Controlled Activation: Enabled
          Delay Before Next Power On: 0.0 seconds
     Hw Address: 46, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
          Shelf Manager Controlled Activation: Enabled
          Delay Before Next Power On: 0.0 seconds
     Hw Address: 47, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
          Shelf Manager Controlled Activation: Enabled
          Delay Before Next Power On: 0.0 seconds
     Hw Address: 48, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
          Shelf Manager Controlled Activation: Enabled
          Delay Before Next Power On: 0.0 seconds
     Hw Address: 49, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
          Shelf Manager Controlled Activation: Enabled
          Delay Before Next Power On: 0.0 seconds
     Hw Address: 4a, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
          Shelf Manager Controlled Activation: Enabled
          Delay Before Next Power On: 0.0 seconds
     Hw Address: 4b, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
          Shelf Manager Controlled Activation: Enabled
          Delay Before Next Power On: 0.0 seconds
     Hw Address: 4c, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
          Shelf Manager Controlled Activation: Enabled
          Delay Before Next Power On: 0.0 seconds
     Hw Address: 4d, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
          Shelf Manager Controlled Activation: Enabled
```

```
        Delay Before Next Power On: 0.0 seconds
    Hw Address: 4e, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds
    Hw Address: 4f, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds
    Hw Address: 50, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds
#
```

# Ändern des Wertes "Maximum External Available Current"

### Syntax:

`shelf maxcurrent` [*zuleitung*] *strom*

### Zweck:

Dieser Befehl legt den Wert für **Maximum External Available Current** (Maximale extern verfügbare Strommenge) für die angegebene Zuleitungsnummer fest und aktualisiert alle bekannten Instanzen der Shelf-FRU-Informationen des Shelfs. Wenn der Parameter *zuleitung* fehlt, wird der Wert für die erste Zuleitung (Feed 0) in den Shelf-FRU-Informationen festgelegt.

Beim Parameter *zuleitung* handelt es sich um eine auf 0 basierende Zuleitungsnummer in den Shelf-FRU-Informationen. Dabei wird die sequenzielle Reihenfolge der Beschreibung einer Zuleitung berücksichtigt.

Beim Parameter *strom* handelt es sich um den aktuellen Soll-Wert in Ampere.

### Beispiel:

Ändern des Wertes für **Maximum Available External Current** für Zuleitung 0 von 44 Ampere in 99 Ampere.

```
# clia shelf pd
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Power Distribution Record (ID=0x11)
```

```
        Version = 0
    Feed count: 1
    Feed 00:
        Maximum External Available Current: 50.0 Amps
        Maximum Internal Current: Not specified
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 2025.000 Watts
        Currently Used Power: 200.000 Watts
        Feed-to-FRU Mapping entries count: 16
           FRU Addr: 41, FRU ID: 0xfe
           FRU Addr: 42, FRU ID: 0xfe
           FRU Addr: 43, FRU ID: 0xfe
           FRU Addr: 44, FRU ID: 0xfe
           FRU Addr: 45, FRU ID: 0xfe
           FRU Addr: 46, FRU ID: 0xfe
           FRU Addr: 47, FRU ID: 0xfe
           FRU Addr: 48, FRU ID: 0xfe
           FRU Addr: 49, FRU ID: 0xfe
           FRU Addr: 4a, FRU ID: 0xfe
           FRU Addr: 4b, FRU ID: 0xfe
           FRU Addr: 4c, FRU ID: 0xfe
           FRU Addr: 4d, FRU ID: 0xfe
           FRU Addr: 4e, FRU ID: 0xfe
           FRU Addr: 4f, FRU ID: 0xfe
           FRU Addr: 50, FRU ID: 0xfe

# clia shelf maxcurrent 0 99
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info
Cached information updated

# clia shelf pd
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Power Distribution Record (ID=0x11)
        Version = 0
    Feed count: 1
    Feed 00:
        Maximum External Available Current: 99.0 Amps
        Maximum Internal Current: Not specified
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 2025.000 Watts
        Currently Used Power: 200.000 Watts
        Feed-to-FRU Mapping entries count: 16
           FRU Addr: 41, FRU ID: 0xfe
           FRU Addr: 42, FRU ID: 0xfe
```

```
            FRU Addr: 43, FRU ID: 0xfe
            FRU Addr: 44, FRU ID: 0xfe
            FRU Addr: 45, FRU ID: 0xfe
            FRU Addr: 46, FRU ID: 0xfe
            FRU Addr: 47, FRU ID: 0xfe
            FRU Addr: 48, FRU ID: 0xfe
            FRU Addr: 49, FRU ID: 0xfe
            FRU Addr: 4a, FRU ID: 0xfe
            FRU Addr: 4b, FRU ID: 0xfe
            FRU Addr: 4c, FRU ID: 0xfe
            FRU Addr: 4d, FRU ID: 0xfe
            FRU Addr: 4e, FRU ID: 0xfe
            FRU Addr: 4f, FRU ID: 0xfe
            FRU Addr: 50, FRU ID: 0xfe
#
```

# Ändern des Wertes von "Minimum Expected Operating Voltage"

### Syntax:

`shelf minvoltage [`*zuleitung*`]` *spannung*

### Zweck:

Dieser Befehl legt den Wert für **Minimum Expected Operating Voltage** (Minimale erwartete Betriebsspannung) für die angegebene Zuleitungsnummer fest und aktualisiert alle bekannten Instanzen der Shelf-FRU-Informationen des Shelfs. Wenn der Parameter *zuleitung* fehlt, wird der Wert für die erste Zuleitung (Feed 0) in den Shelf-FRU-Informationen festgelegt.

Beim Parameter *zuleitung* handelt es sich um eine auf 0 basierende Zuleitungsnummer in den Shelf-FRU-Informationen. Dabei wird die sequenzielle Reihenfolge der Beschreibung einer Zuleitung berücksichtigt.

Beim Parameter *spannung* handelt es sich um den Soll-Wert.

### Beispiel:

Ändern der **Minimum Expected Operating Voltage** für Zuleitung 0.

```
# clia shelf pd
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Power Distribution Record (ID=0x11)
```

```
        Version = 0
    Feed count: 1
    Feed 00:
        Maximum External Available Current: 99.0 Amps
        Maximum Internal Current: Not specified
        Minimum Expected Operating Voltage: -40.5 Volts
        Actual Power Available: 2025.000 Watts
        Currently Used Power: 200.000 Watts
        Feed-to-FRU Mapping entries count: 16
           FRU Addr: 41, FRU ID: 0xfe
           FRU Addr: 42, FRU ID: 0xfe
           FRU Addr: 43, FRU ID: 0xfe
           FRU Addr: 44, FRU ID: 0xfe
           FRU Addr: 45, FRU ID: 0xfe
           FRU Addr: 46, FRU ID: 0xfe
           FRU Addr: 47, FRU ID: 0xfe
           FRU Addr: 48, FRU ID: 0xfe
           FRU Addr: 49, FRU ID: 0xfe
           FRU Addr: 4a, FRU ID: 0xfe
           FRU Addr: 4b, FRU ID: 0xfe
           FRU Addr: 4c, FRU ID: 0xfe
           FRU Addr: 4d, FRU ID: 0xfe
           FRU Addr: 4e, FRU ID: 0xfe
           FRU Addr: 4f, FRU ID: 0xfe
           FRU Addr: 50, FRU ID: 0xfe

# clia shelf minvoltage 0 -59

IPM Sentry Shelf Manager Command Line Interpreter
    Updating Shelf FRU Info
    Cached information updated

# clia shelf pd
IPM Sentry Shelf Manager Command Line Interpreter
    PICMG Shelf Power Distribution Record (ID=0x11)
        Version = 0
    Feed count: 1
    Feed 00:
        Maximum External Available Current: 99.0 Amps
        Maximum Internal Current: Not specified
        Minimum Expected Operating Voltage: -59.0 Volts
        Actual Power Available: 2025.000 Watts
        Currently Used Power: 200.000 Watts
        Feed-to-FRU Mapping entries count: 16
           FRU Addr: 41, FRU ID: 0xfe
```

```
            FRU Addr: 42, FRU ID: 0xfe
            FRU Addr: 43, FRU ID: 0xfe
            FRU Addr: 44, FRU ID: 0xfe
            FRU Addr: 45, FRU ID: 0xfe
            FRU Addr: 46, FRU ID: 0xfe
            FRU Addr: 47, FRU ID: 0xfe
            FRU Addr: 48, FRU ID: 0xfe
            FRU Addr: 49, FRU ID: 0xfe
            FRU Addr: 4a, FRU ID: 0xfe
            FRU Addr: 4b, FRU ID: 0xfe
            FRU Addr: 4c, FRU ID: 0xfe
            FRU Addr: 4d, FRU ID: 0xfe
            FRU Addr: 4e, FRU ID: 0xfe
            FRU Addr: 4f, FRU ID: 0xfe
            FRU Addr: 50, FRU ID: 0xfe
#
```

# Ändern des Flags "Shelf Manager Controlled Activation"

### Syntax:

shelf activation *hardwareadresse fru-id* [1/0]
shelf activation board *n* [1/0]
shelf activation board all [1/0]
shelf activation fan_tray *n* [1/0]

### Zweck:

Dieser Befehl zeigt das Feld **Shelf Manager Controlled Activation** (Vom Shelf Manager gesteuerte Aktivierung) für die angegebene FRU des angegebenen IPM-Controllers an bzw. ändert seinen Inhalt. Der Befehl ändert das Flag **Shelf Manager Controlled Activation** nur für bereits vorhandene Einträge unter **Shelf Activation And Power Management Record** (Datensatz für Shelf-Aktivierung und -Energieverwaltung). Außerdem aktualisiert dieser Befehl die im Cache zwischengespeicherte Version der Shelf-FRU-Informationen, die vom Shelf Manager verwendet werden. Der neue Wert des Felds **Shelf Manager Controlled Activation** wird also sofort wirksam, ohne dass ein Neustart des Shelf Managers erforderlich ist.

Beim Parameter *hardwareadresse* handelt es sich um die 7-Bit-Hardwareadresse im hexadezimalen Format.

Beim Parameter *fru-id* handelt es sich um eine FRU-ID im hexadezimalen Format. 0xFE steht für ALLE FRUs unter dieser Hardwareadresse.

**Beispiel:**

Aktivieren von **Shelf Manager Controlled Activation** für einen IPM-Controller mit der Hardwareadresse 0x42 (IPMB-Adresse 0x84).

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
        Version = 0
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 16
    Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 42, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 43, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 44, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 45, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 46, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 47, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds
```

```
   Hw Address: 48, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

   Hw Address: 49, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

   Hw Address: 4a, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

   Hw Address: 4b, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

   Hw Address: 4c, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

   Hw Address: 4d, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

   Hw Address: 4e, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

   Hw Address: 4f, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

   Hw Address: 50, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds
```

```
# clia shelf activation 42 0xfe 0
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info, address: 0x42, FRU ID # 254
    Cached information updated
    Wrote Information to the Shelf FRU

# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
        Version = 0
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 16
    Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 42, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 43, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 44, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 45, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 46, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 47, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
```

```
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 48, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 49, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 4a, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 4b, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 4c, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 4d, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 4e, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 4f, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 50, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
```

```
        Shelf Manager Controlled Activation: Enabled
        Delay Before Next Power On: 0.0 seconds
#
```

# Ändern des Flags "Shelf Manager Controlled Deactivation"

### Syntax:

shelf deactivation *hardwareadresse fru-id* [1/0]
shelf deactivation board *n* [1/0]
shelf deactivation board all [1/0]
shelf deactivation fan_tray *n* [1/0]

### Zweck:

Dieser Befehl zeigt das Feld **Shelf Manager Controlled Deactivation** (Vom Shelf Manager gesteuerte Deaktivierung) für die angegebene FRU des angegebenen IPM-Controllers an bzw. ändert seinen Inhalt. Der Befehl ändert das Flag **Shelf Manager Controlled Deactivation** nur für bereits vorhandene Einträge im **Shelf Activation And Power Management Record** (Datensatz für Shelf-Aktivierung und -Energieverwaltung). Außerdem aktualisiert dieser Befehl die im Cache zwischengespeicherte Version der Shelf-FRU-Informationen, die vom Shelf Manager verwendet werden. Der neue Wert des Felds **Shelf Manager Controlled Deactivation** wird also sofort wirksam, ohne dass ein Neustart des Shelf Managers erforderlich ist.

Beim Parameter *hardwareadresse* handelt es sich um die 7-Bit-Hardwareadresse im hexadezimalen Format.

Beim Parameter *fru-id* handelt es sich um eine FRU-ID im hexadezimalen Format. 0xFE steht für ALLE FRUs unter dieser Hardwareadresse.

Bei einem Wert von 0 wird **Shelf Manager Controlled Deactivation** *aktiviert*. Dies bedeutet, dass der Shelf Manager für die Deaktivierung des Boards verantwortlich ist. Bei einem Wert von 1 wird **Shelf Manager Controlled Deactivation** *deaktiviert*. Dadurch wird verhindert, dass der Shelf Manager das Board automatisch deaktiviert. Wenn die Option **Shelf Manager Controlled Deactivation** deaktiviert ist, ist der Systemadministrator für die Deaktivierung des Boards verantwortlich.

**Beispiel:**

Aktivieren von **Shelf Manager Controlled Deactivation** für einen IPM-Controller mit der Hardwareadresse 0x41 (IPMB-Adresse 0x84).

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
        Version = 1
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 16
    Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Shelf Manager Controlled Deactivation: Disabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 42, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Shelf Manager Controlled Deactivation: Disabled
        Delay Before Next Power On: 0.0 seconds

# clia shelf deactivation 0x41 0xfe 0
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info, address: 0x41, FRU ID # 254
    Cached information updated
    Wrote Information to the Shelf FRU

# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
PICMG Shelf Activation And Power Management Record (ID=0x12)
        Version = 1
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 16
    Hw Address: 41, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Shelf Manager Controlled Deactivation: Enabled
        Delay Before Next Power On: 0.0 seconds

    Hw Address: 42, FRU ID: 0xfe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Enabled
        Shelf Manager Controlled Deactivation: Disabled
```

```
        Delay Before Next Power On: 0.0 seconds

#
```

# Ändern von "Maximum FRU Power Capability"

### Syntax:

shelf pwrcapability *hardwareadresse fru-id wert*
shelf pwrcapability board *n wert*
shelf pwrcapability fan_tray *n wert*

### Zweck:

Dieser Befehl ändert das Feld **Maximum FRU Power Capability** (Maximale FRU-Leistungskapazität) für die angegebene FRU des angegebenen IPM-Controllers.

---

**Hinweis –** Setzen Sie das Feld **Maximum FRU Power Capability** niemals auf einen Wert, der zu hoch ist und die Sicherheit Ihrer Shelf-Umgebung gefährdet.

---

Der Befehl ändert dieses Feld nur für bereits vorhandene Einträge unter **Shelf Activation And Power Management Record** (Datensatz für Shelf-Aktivierung und -Energieverwaltung). Außerdem aktualisiert dieser Befehl die im Cache zwischengespeicherte Version der Shelf-FRU-Informationen, die vom Shelf Manager verwendet werden. Der neue Wert des Felds **Maximum FRU Power Capability** wird also sofort wirksam, ohne dass ein Neustart des Shelf Managers erforderlich ist.

- Beim Parameter *hardwareadresse* handelt es sich um die 7-Bit-Hardwareadresse im hexadezimalen Format.
- Beim Parameter *fru-id* handelt es sich um eine FRU-ID im hexadezimalen Format. 0xFE steht für ALLE FRUs unter dieser Hardwareadresse.
- Der Parameter *wert* ist der neue Wert des Felds in Watt. Die zulässigen Werte liegen im Bereich 0 bis 65 535.

### Beispiel:

Festlegen von **Maximum FRU Power Capability** für einen IPM-Controller mit der Hardwareadresse 0x42 (IPMB-Adresse 0x84) auf 150 Watt.

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
    Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
```

```
    FRU Activation and Power Description Count: 2
    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds

    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
#
# clia shelf pwrcapability 42 0xfe 150
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info
    Cached information updated
#
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 2
    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds

    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
150 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
#
```

# Ändern von "Delay Before Next Power On"

**Syntax:**

`shelf pwrdelay` *hardwareadresse fru-id wert*
`shelf pwrdelay board` *n wert*
`shelf pwrdelay fan_tray` *n wert*

### Zweck:

Dieser Befehl ändert das Feld **Delay Before Next Power On** für die angegebene FRU des angegebenen IPM-Controllers. Der Befehl ändert dieses Feld nur für bereits vorhandene Einträge unter **Shelf Activation And Power Management Record** (Datensatz für Shelf-Aktivierung und -Energieverwaltung). Außerdem aktualisiert dieser Befehl die im Cache zwischengespeicherte Version der Shelf-FRU-Informationen, die vom Shelf Manager verwendet werden. Der neue Wert des Felds **Delay Before Next Power On** wird also sofort wirksam, ohne dass ein Neustart des Shelf Managers erforderlich ist.

- Beim Parameter *hardwareadresse* handelt es sich um die 7-Bit-Hardwareadresse im hexadezimalen Format.
- Beim Parameter *fru-id* handelt es sich um eine FRU-ID im hexadezimalen Format. 0xFE steht für ALLE FRUs unter dieser Hardwareadresse.
- Der Parameter *wert* ist der neue Wert des Felds in Zehntelsekunden. Die zulässigen Werte liegen im Bereich 0 bis 63.

### Beispiel:

Festlegen von **Delay Before Next Power On** für einen IPM-Controller mit der Hardwareadresse 0x42 (IPMB-Adresse 0x84) auf 5 Sekunden.

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 2
    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
#
# clia shelf pwrdelay 42 0xfe 50
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info
    Cached information updated
#
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
```

```
    FRU Activation and Power Description Count: 2
    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 5.0 seconds
#
```

# Ändern von "Allowance for FRU Activation Readiness"

### Syntax:

`shelf allowance` *wert*

### Zweck:

Dieser Befehl ändert den Parameter **Allowance for FRU Activation Readiness** (Toleranz für FRU-Aktivierungsbereitschaft).

Der Parameter *wert* ist der neue Wert des Parameters in Sekunden. Die zulässigen Werte liegen im Bereich 0 bis 255.

### Beispiel:

Festlegen von **Allowance for FRU Activation Readiness** auf 5 Sekunden.

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 2
    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds

    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
```

```
#
# clia shelf allowance 5

IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info
#
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 5 seconds
    FRU Activation and Power Description Count: 2
    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds

    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
#
```

# Neusortieren der FRU-Aktivierungs- und FRU-Leistungsdeskriptoren

### Syntax:

`shelf pwrreorder` *hardwareadresse-1* *fru-id-1* `before|after` *hardwareadresse-2* *fru-id-2*

Sie können *hardwareadresse fru-id* durch Folgendes ersetzen:

board *n*
fan_tray *n*

### Zweck:

Dieser Befehl ändert die Reihenfolge der FRU-Aktivierungs- und FRU-Leistungsdeskriptoren in den Shelf-FRU-Informationen. Der Befehl kann jedoch nur die bereits vorhandenen Deskriptoren neu sortieren. Außerdem ist es mit der aktuellen Implementierung nur möglich, die Deskriptoren für einen einzelnen **Shelf Activation And Power Management Record** neu zu sortieren. Außerdem aktualisiert dieser Befehl die im Cache zwischengespeicherte Version der Shelf-FRU-Infor-

mationen, die vom Shelf Manager verwendet werden. Die neue Reihenfolge der Deskriptoren wird also sofort wirksam, ohne dass ein Neustart des Shelf Managers erforderlich ist.

- Beim Parameter *hardwareadresse-1* handelt es sich um die hexadezimale 7-Bit-Hardwareadresse des Deskriptors, der verschoben werden soll.
- Beim Parameter *fru-id-1* handelt es sich um die hexadezimale FRU-ID des Deskriptors, der verschoben werden soll. 0xFE steht für ALLE FRUs unter dieser Hardwareadresse.
- Beim Parameter *hardwareadresse-2* handelt es sich um die hexadezimale 7-Bit-Hardwareadresse des Deskriptors, vor bzw. nach dem der Deskriptor *hardwareadresse-1*/`fru`*-id-1* angeordnet werden soll.
- Beim Parameter *fru-id-2* handelt es sich um die hexadezimale FRU-ID des Deskriptors, vor bzw. nach der der Deskriptor *hardwareadresse-1*/`fru`*-id-1* angeordnet werden soll.

### Beispiel:

Anordnen eines Deskriptors für einen IPM-Controller mit Hardwareadresse 0x42 (IPMB-Adresse 0x84) vor dem Deskriptor für einen IPM-Controller mit Hardwareadresse 0x41 (IPMB-Adresse 0x82).

```
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
    FRU Activation and Power Description Count: 2
    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds

    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
#
# clia shelf pwrreorder 42 0xfe before 41 0xfe
IPM Sentry Shelf Manager Command Line Interpreter
Updating Shelf FRU Info
    Cached information updated
#
# clia shelf pm
IPM Sentry Shelf Manager Command Line Interpreter
Power Management:
    Allowance for FRU Activation Readiness: 10 seconds
```

```
    FRU Activation and Power Description Count: 2
    Hw Address: 42, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds

    Hw Address: 41, FRU ID: fe, Maximum FRU Power Capabilities:
200 Watts
        Shelf Manager Controlled Activation: Disabled
        Delay Before Next Power On: 2.2 seconds
#
```

# Aktualisieren der Shelf-FRU-Informationen

### Syntax:

```
shelf info_refresh
```

### Zweck:

Dieser Befehl bewirkt, dass der Shelf Manager die vorher gefundenen Quellen für Shelf-FRU-Informationen im Shelf erneut liest und neu bewertet, welche Quellen gültige Shelf-FRU-Informationen enthalten. Wenn gültige Shelf-FRU-Informationen bestätigt werden, werden alle Speichergeräte für Shelf-FRU-Informationen und die im Cache zwischengespeicherte Masterkopie der Shelf-FRU-Informationen aktualisiert, indem der Inhalt der neuen Shelf-FRU-Informationen verwendet wird.

Der Shelf Manager versucht während der Initialisierung, potenzielle Speichergeräte für Shelf-FRU-Informationen zu finden, wie dies in der Spezifikation PICMG 3.0 (Abschnitt 3.6.4) festgelegt ist. Wenn der Shelf Manager mindestens zwei Geräte mit FRU-Informationen findet, die gültige Shelf-FRU-Informationen enthalten, führt der Shelf Manager eine sog. `election` durch, um die zu verwendenden Quellen für Shelf-FRU-Informationen zu ermitteln. Diese Auswahl basiert auf dem Prüfen der auf den Speichergeräten enthaltenen Daten und dem Vergleichen des Inhalts. Nach einer erfolgreichen Auswahl erstellt der Shelf Manager eine zwischengespeicherte Masterkopie der Shelf-FRU-Informationen (im flüchtigen Speicher), die für alle Aktualisierungen von Shelf-FRU-Informationsquellen verwendet wird und als einzige Quelle für Shelf-FRU-Informationen dient. Auf diese Weise wird für alle Operationen, die sich auf die Shelf-FRU-Informationen beziehen, die Masterkopie verwendet. Außerdem werden alle Änderungen der Masterkopie automatisch als inkrementelle Aktualisierungen auf alle Quellgeräte für Shelf-FRU-Informationen propagiert.

Die dynamische Neukonfiguration wird jedoch nicht unterstützt. Wenn sich die neuen Shelf-FRU-Informationen von den vorherigen Shelf-FRU-Informationen unterscheiden, werden die Änderungen erst vollständig wirksam, nachdem der Shelf Manager neu gestartet wurde.

### Beispiel:

Erfolgreiche Aktualisierung: zwei übereinstimmende Quellen für Shelf-FRU-Informationen.

```
# clia shelf info_refresh

IPM Sentry Shelf Manager Command Line Interpreter

Read 0x20 # 2, size = 1024
Read 0x20 # 1, size = 1024
Found 2 Matching Shelf FRU Info

0x20 # 2, size = 1024 (data size = 775), "Valid" Shelf FRU,
"Matching"
0x20 # 1, size = 1024 (data size = 775), "Valid" Shelf FRU,
"Matching"
Shelf FRU Info was not changed

#


Unsuccessful refresh: both data sources contain non-matching or
invalid data.

# clia shelf info_refresh

IPM Sentry Shelf Manager Command Line Interpreter

Read 0x20 # 2, size = 1024
Read 0x20 # 1, size = 1024
No Matching Shelf FRU Info found

0x20 # 2, size = 1024 (data size = 293), "Invalid" Shelf FRU, "Non-
Matching"
0x20 # 1, size = 1024 (data size = 529), "Valid" Shelf FRU, "Non-
Matching"
Refresh was not done because system found only 1 (of 2) Matching
Shelf FRU info

#
```

## Aktualisieren der Speichergeräte für Shelf-FRU-Informationen

**Syntax:**

`shelf  info_force_update`

**Zweck:**

Dieser Befehl bewirkt eine Prüfung der Speichergeräte für Shelf-FRU-Informationen, und der Inhalt der Masterkopie mit den Shelf-FRU-Informationen wird auf alle Geräte kopiert. Der Befehl ist hilfreich, wenn zwischen der Masterkopie der Shelf-FRU-Informationen und den nicht-flüchtigen Quellgeräten ein Konflikt besteht, der nicht automatisch gelöst werden kann (wenn sich z. B. der Inhalt beider EEPROMs und der Masterkopie unterscheidet). In diesem Fall kann der Bediener die Synchronisierung der EEPROMs mit dem Inhalt der Masterkopie erzwingen, indem er diesen Befehl verwendet. Außerdem löscht dieser Befehl die Fehlerbedingung, die aufgrund des ursprünglichen Konflikts herrscht. Nachdem der Befehl angewendet wurde, werden nachfolgende Aktualisierungen der Shelf-FRU-Informationen auf die SEEPROMs propagiert.

Bei diesem Befehl wird eine asynchrone Aktualisierung der Quellgeräte mit Shelf-FRU-Informationen initiiert.

**Beispiel:**

```
# clia shelf info_force_update

IPM Sentry Shelf Manager Command Line Interpreter

Starting the Shelf FRU Info source device update
#
```

# shelfaddress

**Syntax:**

`shelfaddress  [`*bis_zu_30_zeichen_der_shelf-adresse*`]`

### Zweck:

Dieser Befehl ruft das Feld mit der Shelf-Adresse aus der Adresstabelle in den Shelf-FRU-Informationen ab bzw. legt den Wert dieses Felds fest. Bei dem Befehl werden gepackte 6-Bit-Werte verwendet. Es sind also nur Großbuchstaben und Zahlen zulässig.

Kleinbuchstaben werden automatisch in Großbuchstaben konvertiert.

### Beispiel:

```
# clia shelfaddress
IPM Sentry Shelf Manager Command Line Interpreter
Shelf Address Info: "1234"
#
# clia shelfaddress "NEW SHELF ADDRESS"
IPM Sentry Shelf Manager Command Line Interpreter
Shelf Address Info set successfully
#
# clia shelfaddress
IPM Sentry Shelf Manager Command Line Interpreter
Shelf Address Info: "NEW SHELF ADDRESS"
#
```

## shmstatus

### Syntax:

`shmstatus`

### Zweck:

Dieser Befehl gibt für redundante Konfigurationen den Shelf Manager-Status zurück: **Active** oder **Backup**. Im ausführlichen Modus werden detaillierte Informationen bereitgestellt: Status der Shelf-FRU-Informationen, Status der RMCP-Schnittstelle und Status des Sicherungs-Shelf Managers (wenn es sich beim abgefragten Shelf Manager um den aktiven Shelf Manager handelt). Beim Flag **Ready For Operation** (Betriebsbereit) handelt es sich um einen Parameter, für den in den folgenden Fällen `Yes` angezeigt wird:

- Auf dem aktiven Shelf Manager, wenn gültige Shelf-FRU-Informationen gefunden werden und seine RMCP-Schnittstelle erfolgreich initialisiert wurde.

- Auf dem Sicherungs-Shelf Manager, wenn die Informationen zum Redundanzstatus vom aktiven Shelf Manager erfolgreich empfangen wurden.

**Beispiel:**

```
# clia shmstatus -v
IPM Sentry Shelf Manager Command Line Interpreter
Shelf Manager status: "Active"
Ready For Operation: Yes
Detailed State Flags: "Shelf FRU Found" "RMCP Up" "Backup Healthy"
#
```

# showhost

**Syntax:**

`showhost` *steckplatznummer*

**Zweck:**

Dieser Befehl wird verwendet, um die Versionen der System-Firmware anzuzeigen, die auf einem Netra CP3060 Board installiert sind. Dieser Befehl ist nur für Netra CP3060 Boards gültig, die im Netra CT 900 Server installiert sind.

Der Parameter *steckplatznummer* gibt auf dem Netra CP3060 Board einen Steckplatz an, und die Option *version* wird verwendet, um die vollständigen Versionsinformationen anzuzeigen.

Beispiel:

Anzeigen der aktuellen Firmware-Versionen für das Netra CP3060 Board in Steckplatz 2.

```
# clia showhost 2

System Frimware 6.2.5 Netra CP3060 2006/09/15 15:30

Host flash versions:
 Hypervisor 1.2.3 2006/08/18 12:25
 OBP 4.23.4 2006/08/04 20:46
 Netra[TM] CP3060 POST 4.23.4 2006/08/04 21:17

#
```

# showunhealthy

### Syntax:

showunhealthy

### Zweck:

Dieser Befehl zeigt die Liste der FRUs an, für die ggf. ein Problem vorliegt. Im PICMG 3.0-Kontext enthält diese Liste FRUs, deren Ursache der letzten Hot-Swap-Statusänderung `Communication Lost`, `Communication lost due to local failure` oder `Unexpected deactivation` lautet.

Für jede FRU werden die folgenden Informationen angezeigt: IPMB-Adresse und FRU-Geräte-ID, aktueller Hot-Swap-Status, vorheriger Hot-Swap-Status und Ursache der letzten Statusänderung.

### Beispiel:

Anzeigen der Liste mit Komponenten des Systems, deren Betriebszustand nicht einwandfrei ist.

```
# clia showunhealthy
IPM Sentry Shelf Manager Command Line Interpreter
There are no unhealthy components in the shelf.
#
```

# switchover

### Syntax:

switchover

### Zweck:

Dieser Befehl initiiert den Switchover (Umschaltung) der redundanten Shelf Manager-Instanzen. Sie können den Befehl entweder für die aktive Instanz oder die Sicherungsinstanz des Shelf Managers ausführen.

### Beispiel:

Initiieren des Switchovers (Umschaltung) von der aktiven Instanz bzw. Sicherungsinstanz.

```
# clia switchover
    This Shelf Manager is now active, but is shutting down to
trigger a switchover.
#
```

## terminate

### Syntax:

`terminate`

### Zweck:

Dieser Befehl beendet den Shelf Manager, ohne die Shelf-Verwaltungskarte neu zu starten. Wenn das aktuelle ShMM aktiv ist, wird ein Switchover (Umschaltung) durchgeführt.

### Beispiel:

Beenden des Shelf Managers über die aktive Instanz bzw. Sicherungsinstanz.

```
# clia terminate

#
```

## user

### Syntax:

`user` [*unterbefehl*]

Die folgenden Unterbefehle werden unterstützt:

- `add`
- `delete`

- `enable`
- `name`
- `passwd`
- `channel`

**Zweck:**

Der Befehl `user` zeigt Informationen zu den RMCP-Benutzerkonten des Shelf Managers an und stellt eine einfache Möglichkeit dar, die Benutzerkonten hinzuzufügen, zu löschen und zu ändern.

In den folgenden Unterabschnitten wird die Syntax des Befehls `user` für verschiedene Anwendungsmöglichkeiten beschrieben.

# Anzeigen von Benutzerinformationen

**Syntax:**

`user [-v] [`*benutzernummer*`]`

**Zweck:**

Dieser Befehl zeigt Informationen zu Benutzern an. Wenn Sie den Befehl mit der Option `-v` verwenden, zeigt er auch Informationen zu deaktivierten Benutzern an. (Standardmäßig werden nur aktivierte Benutzer aufgeführt.) Wenn Sie die optionale Benutzernummer angeben, werden nur Informationen zum Benutzer mit der entsprechenden Nummer angezeigt.

Die folgenden Informationen werden angezeigt:

- Benutzernummer
- Benutzername
- Kanalzugriffsinformationen für die einzelnen IPMI-Kanäle: maximale Berechtigungsebene des Benutzers für den Kanal und Kanalzugriffs-Flags

Wenn die Kanalzugriffsinformationen für mehrere Kanäle identisch sind, werden die Daten zusammengefasst und für den Kanalbereich angezeigt.

### Beispiele:

```
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
# clia user -v
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
7: "TEST1" Disabled
        Channels 0-15 Privilege level: "NO ACCESS"
#
```

## Hinzufügen eines neuen Benutzers

### Syntax:

`user add` *benutzernummer benutzername kanalzugriffs-flags berechtigungsebene passwort*

### Zweck:

Dieser Befehl fügt dem System einen neuen Benutzer hinzu. Er stellt gemäß der Angabe im Befehl für alle Kanäle dieselbe maximale Berechtigungsebene und dieselben Kanalzugriffs-Flags ein. Der Befehl gibt einen Fehler zurück, wenn der angegebene Benutzer nicht vorhanden ist. Die Befehlsparameter haben die folgende Bedeutung:

- *benutzernummer* – gültige Benutzernummer
- *benutzername* – Benutzername (wird ohne Meldung auf 16 Zeichen beschränkt)
- *kanalzugriffs-flag* – jeweils das erste Byte des Befehls **SetUserInfo** (nur die Bits 4, 5 und 6 sind aussagekräftig)
  - Bit 6 – IPMI-Messaging aktiviert
  - Bit 5 – Linkauthentifizierung aktiviert
  - Bit 4 – Auf Rückmeldung beschränkt
- *berechtigungsebene* – Berechtigungsebene des Benutzers
- *passwort* – Benutzerpasswort (wird ohne Meldung auf 16 Zeichen beschränkt)

**Beispiel:**

Hinzufügen des Benutzers 9 mit dem Namen `root`, Administrator-Berechtigungsebene und dem Passwort `PICMG guru`.

```
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
# clia user add 9 "root" 0x40 4 "PICMG guru"
IPM Sentry Shelf Manager Command Line Interpreter
User 9 added successfully
#
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
9: "root"
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
```

# Löschen eines Benutzers

**Syntax:**

`user delete` *benutzernummer*

**Zweck:**

Dieser Befehl löscht den Benutzer, der mithilfe der *benutzernummer* angegeben wurde.

**Beispiel:**

Löschen des Benutzers mit der Benutzernummer 10.

```
# clia user delete 10
IPM Sentry Shelf Manager Command Line Interpreter
User 10 deleted successful
#
```

```
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
9: "root"
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
```

## Aktivieren und Deaktivieren eines Benutzers

### Syntax:

`user enable` *benutzernummer* `1 | 0`

### Zweck:

Dieser Befehl aktiviert bzw. deaktiviert einen Benutzer anhand der *benutzernummer*. Der zuletzt angegebene Befehlsparameter steht für die gewünschte Aktion:

- 0 – Angegebenen Benutzer deaktivieren
- Anderer Wert – Angegebenen Benutzer aktivieren

### Beispiele:

Deaktivieren und Aktivieren des Benutzers mit der *benutzernummer* 9.

```
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
9: "root"
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
# clia user enable 9 0
IPM Sentry Shelf Manager Command Line Interpreter
    User 9 disabled successfully
#
# clia user -v
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
```

```
            Flags: "IPMI Messaging"
9: "root" Disabled
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
# clia user enable 9 1
IPM Sentry Shelf Manager Command Line Interpreter
User 9 enabled successfuly
#
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
9: "root"
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
```

# Ändern eines Benutzernamens

### Syntax:

`user name` *benutzernummer benutzername*

### Zweck:

Dieser Befehl wird verwendet, um den Benutzernamen für den angegebenen Benutzer zu ändern. (Der Benutzer wird anhand der Benutzernummer angegeben.) Die Befehlsparameter haben die folgende Bedeutung:

- *benutzernummer* – gültige Benutzernummer
- *benutzername* – Benutzername (wird ohne Meldung auf 16 Zeichen beschränkt)

### Beispiel:

Ändern des Namens von Benutzer 9 in `newby`.

```
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
9: "root"
```

```
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
# clia user name 9 newby
IPM Sentry Shelf Manager Command Line Interpreter
User 9, name changed successfully
#
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
9: "newby"
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
```

# Ändern eines Benutzerpassworts

### Syntax:

`user passwd` *benutzernummer passwort*

### Zweck:

Dieser Befehl wird verwendet, um das Passwort für den angegebenen Benutzer zu ändern. (Der Benutzer wird anhand der Benutzernummer angegeben.) Die Befehlsparameter haben die folgende Bedeutung:

- *benutzernummer* – gültige Benutzernummer
- *passwort* – Benutzerpasswort (wird ohne Meldung auf 16 Zeichen beschränkt)

### Beispiel:

Ändern des Passworts von Benutzer 9 in `RIP`.

```
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
9: "newby"
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
```

```
#
# clia user passwd 9 RIP
IPM Sentry Shelf Manager Command Line Interpreter
User 9, password changed successfully
#
# clia user
IPM Sentry Shelf Manager Command Line Interpreter
1: ""
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
9: "newby"
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
```

# Ändern der Kanalzugriffseinstellungen für einen angegebenen Benutzer und einen angegebenen Kanal

## Syntax:

`user channel` *benutzernummer kanalnummer kanalzugriffs-flags berechtigungsebene*

Zweck: Dieser Befehl wird verwendet, um die Kanalzugriffseinstellung für einen angegebenen Kanal und Benutzer zu ändern. (Der Benutzer wird anhand der Benutzernummer angegeben.) Die Befehlsparameter haben die folgende Bedeutung:

- *benutzernummer* – gültige Benutzernummer
- *kanalnummer* – Kanalnummer
- *kanalzugriffs-flags* – jeweils das erste Byte des Befehls **SetUserInfo** (nur die Bits 4, 5 und 6 sind aussagekräftig)
  - Bit 6 – IPMI-Messaging aktiviert
  - Bit 5 – Linkauthentifizierung aktiviert
  - Bit 4 – Auf Rückmeldung beschränkt
- *berechtigungsebene* – Berechtigungsebene des Benutzers

### Beispiel:

Ändern der maximalen Berechtigungsebene für Benutzer 9 auf Kanal 5 in `User` (Benutzer).

```
# clia user 9
IPM Sentry Shelf Manager Command Line Interpreter
9: "newby"
        Channels 0-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
# clia user channel 9 5 0x60 2
IPM Sentry Shelf Manager Command Line Interpreter
User 9, channel 5 access updated successfully
#
# clia user 9
IPM Sentry Shelf Manager Command Line Interpreter
9: "newby"
        Channels 0-4 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
        Channel 5 Privilege level: "User"
            Flags: "Link Authentication" "IPMI Messaging"
        Channels 6-15 Privilege level: "Administrator"
            Flags: "IPMI Messaging"
#
```

## version

### Syntax:

`version`

### Zweck:

Dieser Befehl zeigt die Versionsinformationen für die Shelf Manager-Software an.

**Beispiel:**

```
# clia version
IPM Sentry Shelf Manager Command Line Interpreter
IPM Sentry Shelf Manager ver. 2.1.3
IPM Sentry is a trademark of Pigeon Point Systems.
Copyright (c) 2002-2005 Pigeon Point Systems
Build date/time: April 3 2006 16:39:37
All rights reserved
#
```

ANHANG **B**

# OEM-IPMI-Befehle von Sun

Die in diesem Anhang beschriebenen Befehle gelten speziell für ATCA Boards, die von Sun Microsystems entwickelt wurden. Die Internet Assigned Numbers Authority (IANA) hat Sun Microsystems die Nummer 42 zugewiesen.

Weitere Informationen zu IANA-Nummernzuweisungen finden Sie unter `http://www.iana.org/assignments/enterprise-numbers`.

Die für diese Befehle verwendete **netfunction** (NetFn) lautet 0x2E. Hierbei handelt es sich um die **OEM netfunction**, die in der IPMI-Spezifikation definiert ist. Dabei müssen die ersten drei Daten-Bytes im Anforderungspaket dieser IANA-Nummer entsprechen, und die ersten drei Bytes, die im Antwortpaket auf den Fertigstellungscode folgen, entsprechen ebenfalls der IANA-Nummer. Bei Sun ATCA Node Boards lauten diese drei Bytes `00 00 2A`.

Die OEM-IPMI-Befehle von Sun sind in TABELLE B-1 aufgeführt und werden in den folgenden Abschnitten beschrieben.

**TABELLE B-1** OEM-IPMI-Befehle von Sun

| Befehl | Opcode | Syntax |
|---|---|---|
| `Get Version` | 0x80 | `#GET_VERSION` |
| `Set Boot Page`† | 0x81 | `#SET_BOOT_PAGE` |
| `Get Boot Page`† | 0x82 | `#GET_BOOT_PAGE` |
| `Set Front panel reset button state` | 0x83 | `#SET_FP_RESET_BUTTON` |
| `Get Front panel reset button state` | 0x84 | `#GET_FP_RESET_BUTTON_STATE` |
| `Set Ethernet Force Front bit` | 0x85 | `#SET_ETH_FORCE_FRONT` |
| `Get Ethernet Force Front bit` | 0x86 | `#GET_ETH_FORCE_FRONT` |
| `Get RTM status` | 0x88 | `#GET_RTM_STATUS` |

† Nur gültig für das Netra CP3010 Node Board

# Get Version

`Get Version` gibt die Firmware-Version und Standby-CPLD-Version des IPM-Controllers (IPMC) zurück. Die Bytes 8, 9 und A sind zur späteren Verwendung reserviert.

| Befehl | NetFn | Opcode | Referenz |
|---|---|---|---|
| `Get Version` | 0x2E (OEM) | 0x80 | - - |

Datenbytes

| Typ | Byte | Datenfeld |
|---|---|---|
| Anforderungs-daten | Byte1 | 00 |
| | Byte2 | 00 |
| | Byte3 | 2A |
| Antwortdaten | Byte1 | Fertigstellungscode:<br>00 = OK<br>C1 = Befehl nicht unterstützt<br>CC = Ungültige Daten in Anforderung<br>(Weitere Fertigstellungscodes finden Sie in der IPMI-Spezifikation.) |
| | Byte2 | 00 |
| | Byte3 | 00 |
| | Byte4 | 2A |
| | Byte5 | CPLD-Version |
| | Byte6 | REV1-Byte der IPMC-Firmware |
| | Byte7 | REV2-Byte der IPMC-Firmware |
| | Byte8 | Für spätere Verwendung reserviert (ignorieren). |
| | Byte9 | Für spätere Verwendung reserviert (ignorieren). |
| | ByteA | Für spätere Verwendung reserviert (ignorieren). |

### *Beispiel (Terminal-Modus):*

```
[B8 00 80 00 00 2A] <-------Anforderung
[BC 00 80 00 00 00 2A 02 02 00 00 00 00] <----Antwort
```

- Die IPMC-Version wird wie folgt gelesen:

  niedrigeres 4-Bit-Wort von REV1 . hohes 4-Bit-Wort von REV2 . niedriges 4-Bit-Wort von REV2

  Im obigen Beispiel lautet die IPMC-Version `2.0.0`

- Die CPLD-Version wird wie folgt gelesen:

  -> niedrigeres 4-Bit-Wort des CPLD-Versions-Byte

  Im Beispiel lautet die CPLD-Version `2`

# Set Boot Page

`Set Boot Page` legt die Startseiten-Bits im Standby-CPLD fest, um die Startseite für Open Boot PROM auszuwählen. Sie können diese Funktion bei Flash-Startfehlern zur Wiederherstellung verwenden. Dieser Befehl gilt nur für das Netra CP3010 Node Board.

| Befehl | NetFn | Opcode | Referenz |
|---|---|---|---|
| `Set Boot Page` | 0x2E (OEM) | 0x81 | CPLD-Spezifikation 1.0 |

Datenbytes

| Typ | Byte | Datenfeld |
|---|---|---|
| Anforderungs-daten | Byte1 | 00 |
| | Byte2 | 00 |
| | Byte3 | 2A |
| | Byte4 | Startseiteneinstellung:<br>Bits 7 bis 2 = 0<br>Bits 1 und 0 = Startseitennummer |
| Antwortdaten | Byte1 | Fertigstellungscode:<br>00 = OK<br>C1 = Befehl nicht unterstützt<br>CC = Ungültige Daten in Anforderung<br>(Weitere Fertigstellungscodes finden Sie in der IPMI-Spezifikation.) |
| | Byte2 | 00 |
| | Byte3 | 00 |
| | Byte4 | 2A |

### *Beispiel (Terminal-Modus):*

```
[B8 00 81 00 00 2A 02] <------Anforderung
[BC 00 81 00 00 00 2A] <------Antwort
```

# Get Boot Page

`Get Boot Page` gibt die aktuellen Einstellungen der ausgewählten Open Boot PROM-Startseite und die aktuellen Einstellungen des Hardware-Switch zurück, der die Startseite auswählt. Dieser Befehl gilt nur für das Netra CP3010 Node Board.

| Befehl | NetFn | Opcode | Referenz |
|---|---|---|---|
| `Get Boot Page` | 0x2E (OEM) | 0x82 | -- |

Datenbytes

| Typ | Byte | Datenfeld |
|---|---|---|
| Anforderungs-daten | Byte1 | 00 |
| | Byte2 | 00 |
| | Byte3 | 2A |
| Antwortdaten | Byte1 | Fertigstellungscode:<br>00 = OK<br>C1 = Befehl nicht unterstützt<br>CC = Ungültige Daten in Anforderung<br>(Weitere Fertigstellungscodes finden Sie in der IPMI-Spezifikation.) |
| | Byte2 | 00 |
| | Byte3 | 00 |
| | Byte4 | 2A |
| | Byte5 | Startseiteneinstellungen<br>Bits 7 bis 4 = Ignorieren. Müssen jeweils 0 sein.<br>Bit 3, 2 = Einstellungen für Hardware-Switch<br>Bit 1, 0 = Momentan eingestellte Startseite |

### *Beispiel (Terminal-Modus):*

```
[B8 00 82 00 00 2A] <------Anforderung
[BC 00 82 00 00 00 2A  02] <------Antwort
```

# Set Front Panel Reset Button State

`Set Front panel reset button state` wird von der Software verwendet, um die Behandlung des Reset-Schalters an der Frontblende durch das CPLD zu ändern, wenn Benutzer diesen Schalter betätigen. Die Standardeinstellung für den CPLD-Start lautet 10.

| Befehl | NetFn | Opcode | Referenz |
|---|---|---|---|
| `Set Front panel reset button state` | 0x2E (OEM) | 0x83 | CPLD-Spezifikation 1.0 |

Datenbytes

| Typ | Byte | Datenfeld |
|---|---|---|
| Anforderungs-daten | Byte1 | 00 |
| | Byte2 | 00 |
| | Byte3 | 2A |
| | Byte4 | Einstellung für Reset-Schalter an der Frontblende.<br>Bits 7 bis 2 = 0<br>Bits 1 und 0 = Status des Schalters an der Vorderseite:<br>00 = IPMC zurücksetzen und POR für CPU bestätigen<br>01 = XIR für CPU<br>10 = POR für CPU<br>11 = Reset-Schalter an Frontblende deaktiviert |
| Antwortdaten | Byte1 | Fertigstellungscode:<br>00 = OK<br>C1 = Befehl nicht unterstützt<br>CC = Ungültige Daten in Anforderung<br>(Weitere Fertigstellungscodes finden Sie in der IPMI-Spezifikation.) |
| | Byte2 | 00 |
| | Byte3 | 00 |
| | Byte4 | 2A |

*Beispiel (Terminal-Modus):*

```
[B8 00 83 00 00 2A 02] <------Anforderung
[BC 00 83 00 00 00 2A] <------Antwort
```

# Get Front Panel Reset Button State

`Get Front panel reset button state` gibt die aktuellen Einstellungen für das Verhalten des Reset-Schalters an der Frontblende zurück. Standardmäßig wird beim CPLD-Start die Einstellung 10 verwendet. Dies bedeutet, dass das Betätigen des Schalters das Zurücksetzen (Power On Reset, POR) der CPU bewirkt.

| Befehl | NetFn | Opcode | Referenz |
|---|---|---|---|
| `Get Front panel reset button state` | 0x2E (OEM) | 0x84 | CPLD-Spezifikation 1.0 |

Datenbytes

| Typ | Byte | Datenfeld |
|---|---|---|
| Anforderungs-daten | Byte1 | 00 |
| | Byte2 | 00 |
| | Byte3 | 2A |
| Antwortdaten | Byte1 | Fertigstellungscode:<br>00 = OK<br>C1 = Befehl nicht unterstützt<br>CC = Ungültige Daten in Anforderung<br>(Weitere Fertigstellungscodes finden Sie in der IPMI-Spezifikation.) |
| | Byte2 | 00 |
| | Byte3 | 00 |
| | Byte4 | 2A |
| | Byte5 | Einstellung für Reset-Schalter an der Frontblende.<br>Bits 7 bis 2 = Nullen<br>Bits 1 und 0 = Status des Schalters an der Frontblende:<br>00 = IPMC zurücksetzen und POR für CPU bestätigen<br>01 = XIR für CPU<br>10 = POR für CPU<br>11 = Reset-Schalter an Frontblende deaktiviert |

*Beispiel (Terminal-Modus):*

```
[B8 00 84 00 00 2A] <------Anforderung
[BC 00 84 00 00 00 2A  02] <------Antwort
```

# Set Ethernet Force Front Bit

Set Ethernet Force Front Bit wird von der Software verwendet, um die Verwendung der Ethernet-Verbindung über das Ein-/Ausgabemodul an der Vorderseite zu erzwingen, wenn die Rear Transition Card im System vorhanden ist. Wenn Sie dieses Bit auf 1 setzen, wird die Ethernet-Verbindung über das Ein-/Ausgabemodul an der Vorderseite erzwungen.

| Befehl | NetFn | Opcode | Referenz |
|---|---|---|---|
| Set Ethernet Force Front Bit | 0x2E (OEM) | 0x85 | CPLD-Spezifikation 1.0 |

Datenbytes

| Typ | Byte | Datenfeld |
|---|---|---|
| Anforderungs-daten | Byte1 | 00 |
| | Byte2 | 00 |
| | Byte3 | 2A |
| | Byte4 | Bit-Einstellung für Ethernet über Vorderseite<br>Bits 7 bis 1 = 0<br>Bits 0 = Vorderseitenstatus erzwingen (1 = Ethernet-Verbindung über Vorderseite erzwingen) |
| Antwortdaten | Byte1 | Fertigstellungscode:<br>00 = OK<br>C1 = Befehl nicht unterstützt<br>CC = Ungültige Daten in Anforderung<br>(Weitere Fertigstellungscodes finden Sie in der IPMI-Spezifikation.) |
| | Byte2 | 00 |
| | Byte3 | 00 |
| | Byte4 | 2A |

### *Beispiel (Terminal-Modus):*

```
[B8 00 85 00 00 2A 01] <------Anforderung
[BC 00 85 00 00 00 2A] <------Antwort
```

# Get Ethernet Force Front Bit

`Get Ethernet Force Front Bit` gibt die aktuelle Einstellung für das Bit zum Erzwingen der Ethernet-Verbindung über die Vorderseite zurück.

| Befehl | NetFn | Opcode | Referenz |
|---|---|---|---|
| `Get Ethernet Force Front Bit` | 0x2E (OEM) | 0x86 | CPLD-Spezifikation 1.0 |

Datenbytes

| Typ | Byte | Datenfeld |
|---|---|---|
| Anforderungs-daten | Byte1 | 00 |
| | Byte2 | 00 |
| | Byte3 | 2A |
| Antwortdaten | Byte1 | Fertigstellungscode:<br>00 = OK<br>C1 = Befehl nicht unterstützt<br>CC = Ungültige Daten in Anforderung<br>(Weitere Fertigstellungscodes finden Sie in der IPMI-Spezifikation.) |
| | Byte2 | 00 |
| | Byte3 | 00 |
| | Byte4 | 2A |
| | Byte5 | Bit-Einstellung für Ethernet über Vorderseite<br>Bits 7 bis 1 = 0<br>Bit 0 = Bit-Einstellung für Ethernet über Vorderseite (1 = Ethernet-Verbindung über Vorderseite erzwingen) |

## *Beispiel (Terminal-Modus):*

```
[B8 00 86 00 00 2A] <------Anforderung
[BC 00 86 00 00 00 2A 01] <------Antwort
```

# Get RTM Status

Sie können `Get RTM Status` verwenden, um das Vorhandensein einer RTC (Rear Transition Card) im System zu erkennen.

| **Befehl** | **NetFn** | **Opcode** | **Referenz** |
|---|---|---|---|
| `Get RTM Status` | 0x2E (OEM) | 0x88 | CPLD-Spezifikation |

Datenbytes

| **Typ** | **Byte** | **Datenfeld** |
|---|---|---|
| Anforderungs-daten | Byte1 | 00 |
| | Byte2 | 00 |
| | Byte3 | 2A |
| Antwortdaten | Byte1 | Fertigstellungscode:<br>00 = OK<br>C1 = Befehl nicht unterstützt<br>CC = Ungültige Daten in Anforderung |
| | Byte2 | 00 |
| | Byte3 | 00 |
| | Byte4 | 2A |
| | Byte5 | Vorhandensein der RTC<br>Bits 7 bis 1 = 0<br>Bit 0 = Vorhandensein der RTC (0 = RTC nicht erkannt, 1 = RTC erkannt) |

## *Beispiel (Terminal-Modus):*

```
[B8 00 88 00 00 2A] <------Anforderung
[BC 00 88 00 00 00 2A 01] <------Antwort
```

# Glossar

Als Verwalter des Netra CT 900 Servers sollten Sie die Bedeutung der folgenden Begriffe und Abkürzungen kennen.

## A

**ATCA** Abkürzung für Advanced Telecom Computing Architecture; auch als AdvancedTCA bezeichnet. Eine Reihe von Industriestandard-Spezifikationen für die nächste Generation von Carrier Grade-Kommunikationsservern. AdvancedTCA vereint die neuesten Trends in superschnellen Interconnect-Technologien und Prozessoren der nächsten Generation mit verbesserter Zuverlässigkeit, Bedienbarkeit und Reparaturfreundlichkeit. Neue, optimierte Blade- (Platinen) und Gehäuse-(Shelf)-Formate wurden entwickelt, die aufgrund der Standardisierung die Kommunikation auf niedrigem Kostenniveau ermöglichen.

## B

**Basiskanal** Eine physikalische Verbindung innerhalb der Basisschnittstelle, die sich aus bis zu vier Differenzsignalpaaren zusammensetzt. Dabei ist jeder Basiskanal der Endpunkt einer Verbindung zwischen zwei Steckplätzen innerhalb der Basisschnittstelle.

**Basisschnittstelle** Eine Schnittstelle, die 10/100- oder 1000BASE-T-Verbindungen zwischen Node Boards und Switches in einem Shelf unterstützt. Zur Unterstützung der Basisschnittstelle bedarf es Mittelplatinen. Diese senden vier

Differenzsignalpaare zwischen den Node Board-Steckplätzen und den Switch-Steckplätzen (Steckplätze für Basis-Switch im Netra CT 900 Server sind die physikalischen Steckplätze 7 und 8, logische Steckplätze 1 und 2).

**Basis-Switch** Ein Switch zur Unterstützung der Basisschnittstelle. Ein Basis-Switch gewährleistet 10/100/1000BASE-T-Paketvermittlungsdienste zu allen im Shelf installierten Node Boards. Die Basis-Switches im Netra CT 900 Server befinden sich in den physikalischen Steckplätzen 7 und 8 (logische Steckplätze 1 und 2) des Shelfs und unterstützen die Verbindungen zu allen Node Boards und Knoten-Steckplätzen. Boards zur Unterstützung von Fabric- und Basisschnittstellen werden ebenso als „Switches“ bezeichnet.

## D

**Datentransport-schnittstelle** Eine Reihe von Punkt-zu-Punkt-Schnittstellen und Bus-geleiteten Signalen, die eine Verbindung zwischen der Datenlast an Switches und Node Boards herstellen.

**Dual Star-Topologie** Eine Fabric-Verbindungs-Topologie, in der zwei Switches redundante Verbindungen zu allen Endpunkten im Netzwerk bereitstellen. Ein Switch-Paar bietet redundante Verbindungen zwischen Node Boards.

## E

**Elektronisches Keying oder E-Keying** Ein Protokoll, das die Kompatibilität der Verbindungen zwischen der Basisschnittstelle, der Fabric-Schnittstelle, der Update-Kanal-Schnittstelle und dem Synchronisierungstakt der Frontplatine beschreibt.

**ETSI** European Telecommunications Standards Institute.

## F

**Fabric-fähige Platine** Eine Platine, die Verbindungen zu allen anderen Platinen der Mittelplatine bereitstellt. Fabric-fähige Platinen unterstützen die Fabric-Schnittstelle und können auch die Basisschnittstelle unterstützen. Sie können über 2 bis 15

Fabric-Schnittstellenkanäle (meist über alle 15 Kanäle) direkte Verbindungen zu den anderen Platinen im Shelf herstellen. Durch die Anzahl der unterstützten Kanäle wird die maximale Anzahl der Platinen festgelegt, die in einem Shelf verbunden werden können. Fabric-fähige Platinen ohne Verwendung der Basisschnittstelle können im niedrigsten der verfügbaren logischen Steckplätze installiert werden, solche mit Unterstützung der Basisschnittstelle können als Basis-Switches fungieren. Sie können dann die Basiskanäle 1 und 2 unterstützen und in die logischen Steckplätze 3 bis 16 installiert werden. Platinen mit Unterstützung der Basisschnittstelle verwenden die Basiskanäle 1 und 2 nur zur Unterstützung von 10/100/1000BASE-T Ethernet.

**Fabric-Kanal** Ein Fabric-Kanal besteht aus zwei Signalpaarreihen für insgesamt acht Signalpaare pro Kanal. Somit unterstützt jeder einzelne Anschluss bis zu fünf Kanäle für Punkt-zu-Punkt-Verbindungen. Ein Kanal kann aber auch als ein Konstrukt aus vier 2-paarigen Ports betrachtet werden.

**Fabric-Schnittstelle** Eine Zone-2-Schnittstelle, die 15 Verbindungen pro Platine bzw. pro Steckplatz bietet, wobei jede Verbindung bis zu acht Differenzsignalpaare (Kanäle) umfasst und Verbindungen mit bis zu 15 anderen Steckplätzen oder Platinen unterstützt. Mittelplatinen können die Fabric-Schnittstelle in einer Vielzahl von Konfigurationen, einschließlich vollständig vermaschte und Dual Star-Topologien, unterstützen. Platinen, die die Fabric-Schnittstelle unterstützen, können als Fabric Node Boards, Fabric-Switches oder Fabric-fähige Platinen konfiguriert werden. Implementierungen für Fabric-Schnittstellenplatinen werden in den untergeordneten Spezifikationen zu PICMG 3.x definiert.

**Field Replaceable Unit (FRU)** Aus der Servicesicht sind das die kleinsten, nicht weiter zerlegbaren Elemente eines Servers. Beispiele für FRUs sind Festplattenlaufwerke, I/O-Karten und Stromversorgungseinheiten. Ein Server mit all seinen Karten und anderen Komponenten ist keine FRU. Ein leerer Server (ohne Karten und Komponenten) ist dagegen eine FRU.

**Frontplatine** Eine Platine, die den mechanischen Spezifikationen der PICMG 3.0 (8U x 280 mm) entspricht und sich aus der Leiterplatte und einem Bedienfeld zusammensetzt. Die Frontplatine verbindet die Mittelplatinenanschlüsse der Zone 1 mit denen der Zone 2. Sie kann optional auch Verbindungen zu einem Mittelplatinenanschluss der Zone 3 oder direkt zum RTC-Anschluss herstellen und wird an der Vorderseite des Shelfs installiert.

## H

**Hot-Swap** Verfahren zur Verbindung und Trennung von Peripheriegeräten oder anderen Komponenten bei laufendem Systembetrieb. Dieses Leistungsmerkmal setzt eine bestimmte Gestaltung der Hard- und Software voraus.

# I

**$I^2C$** Inter-Integrated Circuit-Bus. Ein vielseitiger, serieller Zweidraht-Bus, der als Basis für aktuelle IPMBs dient.

**IPMB** (Intelligent Platform Management Bus) Der Management-Bus auf unterster Hardware-Ebene, der in den Spezifikationen des Intelligent Platform Management Bus Communications Protocol beschrieben wird.

**IPMB-0-Hub** Ein Hub-Gerät, das mehrere sternförmige IPMB-0-Links zu unterschiedlichen FRUs im System bereitstellt. So ist ein IPMB-0-Hub-Gerät z. B. in einem ShMC mit sternförmigen IPMB-0-Links integriert.

**IPMB-0-Link** Entspricht in einer sternförmigen Topologie dem physikalischen IPMB-0-Segment zwischen einem IPMB-0-Hub-Gerät und einer einzelnen FRU. Dabei ist jeder IPMB-0-Link in einem IPMB-0-Hub-Gerät meist mit einem eigenen IPMB-0-Sensor verknüpft. Er dient in einer Bus-Topologie auch als Verbindung zu mehreren FRUs.

**IPM-Controller (IPMC)** Der Teil einer FRU, der die Verbindung zum ATCA IPMB-0 realisiert und dort die entsprechende FRU und eventuell vorhandene Nebengeräte darstellt.

**IPMI** (Intelligent Platform Management Interface) Eine Spezifikation und Methode zur Bereitstellung von Bestandsverwaltung, Überwachung, Protokollierung und Steuerung der Elemente eines Computersystems, die in der Intelligent Platform Management Interface-Spezifikation beschrieben werden.

# K

**Kartensteckplatz** Ein Steckplatz auf der Mittelplatine, der ausschließlich Node Boards unterstützt. Ein Kartensteckplatz kann keine Switches unterstützen, daher kann ein Node Board in keinem Fall die logischen Steckplätze 1 und 2 belegen. Kartensteckplätze dienen nur für Mittelplatinen, die für sternförmige Topologien ausgelegt sind. Sie unterstützen sowohl die Basis- als auch Fabric-Schnittstelle, meist zwei oder vier Fabric-Kanäle und die Basiskanäle 1 und 2. Jeder Kartensteckplatz mit zwei Kanälen stellt jeweils Verbindungen zu den logischen Steckplätzen 1 und 2 her. Kartensteckplätze mit vier Kanälen stellen jeweils Verbindungen zu den logischen Steckplätzen 1, 2, 3 und 4 her.

# L

**Logic Ground (Signalmasse)** Ein shelfweites elektrisches Netz, das an Platinen und Mittelplatinen als eine Referenz- und Rückgabeleitung für logische Signale dient, die zwischen den Platinen übertragen werden.

# M

**Mittelplatine (Midplane)** Die Mittelplatine (Midplane) ist von ihrer Funktion her das Äquivalent zu einer Rückwandplatine (Backplane) und ist an der Rückseite des Servers befestigt. Die CPU-Karte, Speichergeräte und I/O-Karten werden von der Vorderseite des Gehäuses aus in die Mittelplatine gesteckt, während die RTC-Module von der Rückseite des Gehäuses aus mit der Mittelplatine verbunden werden.

# N

**NEBS** (Network Equipment/Building System) Eine Reihe von Anforderungen, die an in US-Telekommunikationskontrollzentralen installierte Ausrüstung gestellt wird. Dazu gehören Regeln und Vorschriften zum Arbeitsschutz, zum Schutz des Eigentums und zur Aufrechterhaltung des Betriebs. Für die NEBS-Tests wird die Ausrüstung unterschiedlichen Schwingungsbelastungen, Feuer und anderen Umwelt- und Qualitätsfaktoren ausgesetzt. Es gibt drei Stufen für die NEBS-Zertifizierung, die aufeinander aufbauen. Ausrüstungsteile, die die Vorschriften der NEBS-Stufe 3 erfüllen, können sicher unter den Bedingungen einer „extremen Umwelt“ eingesetzt werden. Dabei werden Telekommunikationszentralen als „extreme Umwelt“ eingestuft.

Verantwortlich für die Festlegung der NEBS-Standards ist die Firma Telcordia Technologies, Inc., vormals Bellcore.

**Node Board** Eine Platine für den Einsatz in einer sternförmig verschalteten Mittelplatine, die Verbindung zu einem in der Mittelplatine integrierten Switch bereitstellt. Node Boards können sowohl die Basis- wie auch die Fabric-Schnittstelle unterstützen. Node Boards zur Unterstützung der Fabric-Schnittstelle verwenden Fabric-Kanäle 1 und 2, Boards zur Unterstützung der Basisschnittstelle verwenden die Basiskanäle 1 und 2 nur für 10/100/1000BASE-T Ethernet.

## P

**PCI** (Peripheral Component Interconnect) Ein Standard für Verbindungen zur Peripherie eines Computers mit einer Taktfrequenz von 20 - 33 MHz und einer Datenbreite von 32 Bit über einen 124-poligen Anschluss oder von 64 Bit über einen 188-poligen Anschluss. Pro Takt wird eine Adresse gefolgt von einer Datenbreite (oder mehreren im Burst-Modus) gesendet.

Technisch gesehen handelt es sich bei PCI nicht um einen Bus, sondern vielmehr um eine Brücke oder ein Mezzanine, das durch den integrierten Pufferspeicher die CPU von relativ langsamen Peripheriegeräten entkoppelt und diesen einen asynchronen Betrieb ermöglicht.

**Physikalische Adresse** Eine Adresse, mit der die Lage des physikalischen Steckplatzes einer FRU definiert wird. Sie setzt sich aus einer Kennung des Standorttyps und der Standortnummer zusammen.

**PICMG** (PCI Industrial Computer Manufacturers Group) Ein Konsortium von Unternehmen, das offene Spezifikationen für Computeranwendungen für die Telekommunikation und Industrie entwickelt, z. B. den Standard CompactPCI.

## R

**Rahmen** Eine physikalische oder logische Einheit, die ein oder mehrere Shelves enthalten kann. Sie wird auch als Rack oder, als Einbauvariante, als Schrank bezeichnet.

**RAS (Reliability, Availability, Serviceability)** (dt.: Zuverlässigkeit, Verfügbarkeit und Reparaturfreundlichkeit) Hardware- und Softwarefunktionen bzw. -merkmale, die Zuverlässigkeit, Verfügbarkeit und Reparaturfreundlichkeit eines Servers gewährleisten oder verbessern.

**Rear Transition Card** Ein Karte, die nur bei Netra CT 900 Server-Modellen mit Zugang von der Rückseite eingesetzt wird, um die Anschlüsse bis zur Rückseite des Shelfs zu verlängern.

# S

**Shelf** Eine Komponentenbaugruppe, die aus Mittelplatine, Frontplatinen, Lüftern, Rear Transition Cards und Stromversorgungsmodulen besteht. Das Shelf wurde früher als Chassis bezeichnet.

**Shelf-Adresse** Ein Deskriptor von variablem Format und variabler Länge von bis zu 20 B, der jedes Shelf innerhalb der Verwaltungsdomäne (Management Domain) kennzeichnet.

**Shelf Manager** Die Einheit im System, die in einem AdvancedTCA-Shelf die Stromversorgung, Lüftung und Verbindungen (mit Electronic Keying) bereitstellt. Der Shelf Manager sendet Meldungen zwischen System Manager-Schnittstelle und IPMB-0, bietet Schnittstellen zu System-Repositorys und reagiert auf Ereignismeldungen. Der Shelf Manager kann teilweise oder vollständig auf dem ShMC oder der System Manager-Hardware ausgeführt werden.

**Shelf-Masse** Eine mit dem Rahmen verbundene Schutzerdung und Masserückleitung, die für alle Platinen verfügbar ist.

**ShMC** (Shelf Management Controller) Ein IPM-Controller, der auch Funktionen des Shelf Managers unterstützen kann.

**Sicherungs-Shelf-Verwaltungskarte** Beliebige Shelf-Verwaltungskarte, die für die Shelf-Verwaltungsfunktion Support-Funktionen übernehmen kann.

**SNMP** Abkürzung für Simple Network Management Protocol.

**Sterntopologie** Eine Mittelplatinen-Topologie mit einem oder mehreren Hub-Steckplätzen, die die Verbindungen zwischen den unterstützten Knotensteckplätzen bereitstellt.

**Switch** Eine Platine für den Einsatz in einer sternförmig verschalteten Mittelplatine, die Verbindungen zu einer Anzahl von in der Mittelplatine integrierten Node Boards bereitstellt. Switches können entweder die Basis- oder die Fabric-Schnittstelle oder beide unterstützen. Platinen, die die Fabric-Schnittstelle verwenden, bieten in der Regel Schaltressourcen zu allen 15 Fabric-Kanälen. Switches mit Unterstützung der Basisschnittstelle werden in den logischen Steckplätzen 1 und 2 installiert und verwenden alle 16 Basiskanäle zur Bereitstellung von 10/100/1000BASE-T Ethernet-Schaltressourcen zu bis zu 14 Node Boards und zum anderen Switch. Ein Basiskanal wird zur Unterstützung einer Verbindung zur Shelf-Verwaltungskarte zugewiesen.

**Switch-Steckplatz** In einer sternförmig verschalteten Mittelplatine müssen Switch-Steckplätze die logischen Steckplätze 1 und 2 belegen. Sie unterstützen sowohl die Basis- wie auch die Fabric-Schnittstelle. In den logischen Steckplätzen 1 und 2 installierte Switch-Steckplätze können Switches für die Basis- und die Fabric-Schnittstelle

unterstützen. Die logischen Steckplätze 1 und 2 sind unabhängig von der Fabric-Topologie nur für Switches ausgelegt und unterstützen jeweils bis zu 16 Basiskanäle und bis zu 15 Fabric-Kanäle.

**System** Eine verwaltete Einheit, die eine oder mehrere der folgenden Komponenten enthält: Knoten und Switches, Shelves und Rahmen.

# U

**U** Eine Maßeinheit, die 1,75 Zoll (4,45 mm) entspricht.

**Update-Kanal-Schnittstelle** Eine Zonen-2-Schnittstelle, die (auch als Update-Kanal bezeichnet) zwischen Platinen Verbindungen bereitstellt, die sich aus zehn Differenzsignalpaaren zusammensetzen. Anhand dieser direkten Verbindung zwischen zwei Platinen lassen sich Statusinformationen synchronisieren. Der für den Update-Kanal einer Platine implementierte Transport ist nicht definiert. Über einen Update-Kanal lassen sich jedoch nur zwei Platinen mit gleicher Funktion von einem Hersteller verbinden. Anhand von Electronic Keying wird sichergestellt, dass die Endpunkte eines Update-Kanals übereinstimmende Transportprotokolle verwenden, die vor der Aktivierung der Treiber zugewiesen werden. Mittelplatinen müssen den Update-Kanal unterstützen, auf den Platinen ist die Unterstützung optional.

# V

**Vollständiger Kanal** Eine Fabric-Kanalverbindung, die alle acht Differenzsignalpaare zwischen den Endpunkten nutzt.

**Vollständig vermaschte Topologie (Full Mesh)** Eine vollständig vermaschte Konfiguration, die in der Fabric-Schnittstelle verwendet wird, um einen dedizierten Verbindungskanal zwischen den einzelnen Steckplatzpaaren im Shelf herzustellen. Mittelplatinen mit der Full Mesh-Konfiguration können Fabric-fähige Platinen und Node Boards in einer Dual Star-Anordnung unterstützen.

# Z

**Zone 1** Die schmale Fläche am oberen/unteren Rand eines ATCA-Steckplatzes, die für Stromversorgung, Verwaltungs- und andere Hilfsfunktionen vorgesehen ist.

**Zone 2** Die schmale Fläche am oberen/unteren Rand eines ATCA-Steckplatzes, die für die Schnittstelle zum Datentransport vorgesehen ist.

**Zone 3** Die schmale Fläche am oberen/unteren Rand eines ATCA-Steckplatzes, die für benutzerdefinierte Verbindungen und/oder Verbindungen zu den Rear Transition Cards für Systeme mit Zugriff von der Rückseite reserviert ist.

**Zugang von der Rückseite** Eine Konfigurationsoption für den Netra CT 900 Server, wobei alle Kabel auf die Rückseite des Shelfs geführt werden.

# Index

## T



## U



## W



## Z